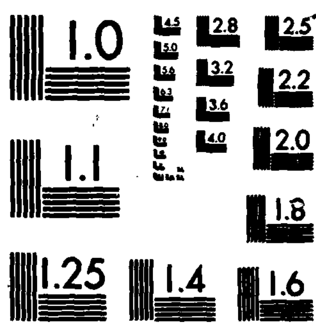
AD-A163 530 DSCS NETWORK PERFORMANCE SOFTWARE SUPPORT REVISION(U) 1/2
M/A-COM LINKABIT INC VIENNA VA S V TREES ET AL.
07 JAN 86 DCA100-84-C-0009
UNCLASSIFIED F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

M/ACOM
GOVERNMENT SYSTEMS DIVISION
8619 WESTWOOD CENTER DRIVE
VIENNA, VA 22180

M/A-COM LINKABIT, Inc.
Log # MSO-85-093b
Copy # 13

2

DSCS NETWORK PERFORMANCE SOFTWARE SUPPORT
(Revised)

Final Report 170
Task MSO85-5

January 1986

Prepared by M/A-COM LINKABIT, Inc.
Under Contract DCA100-84-C-0009

Submitted to
Defense Communications Agency
Center for Command and Control,
and Communications Sysems, Code A800
8th & S. Courthouse Road
Arlington, VA 22204

511270.1

CENTER FOR COMMAND AND CONTROL, AND COMMUNICATIONS SYSTEMS (C^4S)

" EXCELLENCE IN C^3 SYSTEMS FOR NATIONAL DEFENSE"

DSCS NETWORK PERFORMANCE SOFTWARE SUPPORT

Final Report
January 1986

Accesion For	
NTIS CRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

DEFENSE COMMUNICATIONS AGENCY

QUALITY INSPECTED 3

C^4S ______

CY ______ OF ______ CYS

511271.0

Table of Contents

List of Figures

List of Tables

EXECUTIVE SUMMARY

This report documents the results of software enhancement and version release test work for the DSCS Network Performance System (DNPS). The effort was addressed through three subtasks:

- Subtask A--Requirements Analysis, Development, and Test
- Subtask B--Version Release Test Support
- Subtask C--Integration of the Receive Multiple Beam Antenna (MBR) Resource Allocator.

The subtasks were conducted through the period December 1984 - November 1985.

Subtask A - Requirements Analysis, Development, and Test

The purpose of Subtask A was to evaluate the utilization of the DSCS Operational Control System (DOCS). Although the results of this work are detailed in a task Interim Report published on 9 Sept 1985 (repeated here in Section B.1 through B.4 of Appendix B for convenience), two new recommendations for enhancements to the DNPS are contained in Chapter 1 of this report.

Subtask B - Version Release Test Support

The purpose of Subtask B was to maintain software integrity of the DOCS. Efforts included testing of the DNPS version 3.1 electronic counter-countermeasure (ECCM) subsystem and verification of the DSCS Operational Support System (DOSS) 2.1 and DNPS 3.1 Allocation and Performance subsystems. An overview of this testing effort is provided in Chapter 2. Appendix A contains the Test Plan/Report.

Subtask C - Integration of the Receive Multiple Beam Antenna (MBR) Resource Allocator

Under Subtask C, a previously developed multiple beam antenna (MBA) resource allocator algorithm (software name, MBR_AAS) was integrated into the DNPS under the analysis support function. Results obtained with the MBR_AAS software exhibited slow convergence to the desired MBA gain pattern and, in many cases, an inability to converge

to an acceptable gain pattern. Efforts therefore focused on the development of a new MBR steering algorithm with improved convergence properties. Initial results of the new algorithm, the Cholesky reduction algorithm (lower/upper triangular matrix decompostion), are in excellent agreement with the user-supplied MBR (desired) directive gain contours. An evaluation of the Cholesky reduction algorithm and results of its application in this work are provided in Chapter 3. This report provides a detailed examination of the MBR_AAS algorithm in Sections B.5 through B.7 of Appendix B, with test results in Appendix C.

Chapter 1

REQUIREMENTS ANALYSIS, DEVELOPMENT, AND TEST

1.1 SUBTASK A

The purpose of this subtask was to develop and implement modifications of the DCEC version of the DSCS Operational Control System (DOCS) Network Planning Software (DNPS) in response to needs identified by DCEC.

In concert with the DCEC task officer, it was agreed that the majority of the effort expended under this subtask was to be an examination of the current DOCS/DNPS structure and general techniques that could be used for growth and expansion of capabilities. Most efforts on this subtask were completed in September 85 and presented in an interim report. For ease of reference, the earlier reported results of Subtask A are given herein in Appendix B.

The top-level functional organization of DNPS is given in Figure 1-1. The functions of DNPS use a complex series of operator inputs and runtime definitions to compute the resultant evaluation of DSCS network performance. The following additional recommendations would assist the operator and enhance the software:

- **DNPS Interactive Help Function -**
  Using the 'Utilities' section of the program as the entry point, the help function would echo the top-level DNPS menu. Help is displayed on the terminal when the operator selects a subsystem (for example '2,' which is currently the procedures subsystem). The program could then offer assistance on the operator's terminal ranging from a terse overview to a very detailed description of the subsystem. The data displayed would be determined by the operator or the "level" at which this enhancement is implemented. The "level" of enhancement is the level at which the DNPS program is documented (via the enhancement). For example, the Scenario Definition subsystem offers many sublevels of program usage, thus the level of enhancement is the point/sublevel that the documentation stops. The following improvements could be realized:

  - Improved operator performance -
    The user could have program documentation displayed when needed. The display could be copied to the printer if desired.

Figure 1-1. Functional Organization of DNPS

```
                 +----------------------+
                 |                      |
                 |  DNPS   Supervisor   |
                 |                      |
                 +----------+-----------+
                            |
+---------------------------+
|
+---------+-------------+-------------+-------------+
|         |             |             |             |
|  +------v------+ +-------v------+ +------v------+ +-----v------+
|  |  Procedure  | | Static Data  | |  Scenario   | |  Analysis  |
|  | Definitions | | Base Loading | | Definitions | |  Support   |
|  +-------------+ +--------------+ +-------------+ +------------+
|
+---------+-------------+-------------+-------------+
|         |             |             |             |
|  +------v------+ +-------v------+ +------v------+ +-----v------+
|  | Satellite   | | Net. Resource| |Net. Perform.| |   Report   |
|  |  Loading    | |  Allocation  | |  Analysis   | | Generation |
|  +-------------+ +--------------+ +-------------+ +------------+
|
+---------+-------------+-------------+-------------+
          |             |             |             |
   +------v------+ +-------v------+ +------v------+ +-----v------+
   |   ECCM      | |     DASA     | | Sat. Tasking| |  Special   |
   |  Adaptation | |              | |  & Config.  | |Requirements|
   +-------------+ +--------------+ +-------------+ +------------+
```

Up-to-date manual -
As the program changes, the documentation should be concurrently updated/refined.

- Program limitations -
Any known problems and/or limitations of the current version should be listed. This would avoid the issuance of redundant Software User Reports (SURs).

- **Improved User Procedures -**
The ability to 'pass' an argument to a procedure as it is invoked would enhance the program. This argument would be a datum required in the body of the procedure and be substituted at the point it is used. The definition of the argument (identification of the argument to the program) is determined at the time the procedure is created. An alternate/additional method is to terminate input from the procedure file and obtain input from the operator. The procedure would then continue reading data from the procedure file.

Data modification provides an example of how this enhancement could be used. *In a sample invocation, the data rate is the user-defined argument.* The body of the procedure would step through the program subsystems to change the data rate of various links; then, reports would be generated. The procedure would be invoked again and a different data rate supplied and reports generated. The time required to perform the change and produce reports would be significantly reduced. The creation of one procedure would speed the ability of an operator to evaluate the effects of many different data rates.

PREVIOUS PAGE IS BLANK

Chapter 2

VERSION RELEASE TEST SUPPORT

2.1 SUBTASK B

As modifications were made or were planned to be made to the operationally fielded system, these changes were incorporated into the DCEC system, and the resultant software was tested for performance integrity. During this modification and test process, an exact duplicate of the operational software was maintained for use by DCEC engineers.

2.1.1 Test Support Overview

Testing and analysis was performed under this subtask on the 3.1 version of DNPS. In addition, regression testing was performed on version 2.1 of DNPS using a typical predefined scenario. This regression testing was performed to ensure that any major changes which were made to the computational algorithms provided validated output predictions.

The preliminary Version Test Plan/Report is given in Appendix A. The analysis phase of the testing effort is still in progress. The final version of the Test Plan and Report is scheduled for release in 1986.

2.1.2 Test Support Recommendations

Because of the numerous submenus of DNPS and menus that are only displayed when certain conditions arise, the following recommendations are made:

- Test procedures should be provided through a single source.
- Concise and approved test procedures should be provided with any release of the software.
- The test procedures should be addressed at the time any software modifications are introduced at the Configuration Control Board review. The level of testing required for acceptance should be resolved when the change is accepted.
- The DNPS package should be tested using automated methods. This would ensure that any previously working code remains unaffected and any new code will be tested. Automated testing would create known output and

program response (based on automated input). Deviations in program response and output (in a new version of software) would indicate a change in the software package.

Chapter 3

INTEGRATION OF THE RECEIVE MULTIPLE BEAM ANTENNA RESOURCE ALLOCATOR

3.1 SUBTASK C

DCEC personnel have begun work on an algorithm to optimize the allocation of the DSCS III receive multiple beam antenna and to allow for the definition of "areas" of antenna coverage and directive gain. This algorithm currently resides on the VAX-11/780 computer at the DCEC Hybrid Simulation Facility. Under this subtask, work on the development of the optimization algorithm continued (to include a demonstration test conducted for the government) to the point where the algorithm could be integrated into the DNPS.

Analysis of the MBR_AAS algorithm was provided in an interim task report and is given in Appendix B (Sections B.5 through B.7) and in Appendix C for ease of reference.

3.2 SUMMARY OF NEW BEAM FORMING ALGORITHM STUDIES

The MBR antenna is composed of 61 individual singlet beams whose outputs are electronically (using in-phase and quadrature weights at RF) weighted and summed to produce a desired composite gain distribution on the earth's surface.

Two numerical analysis techniques have been investigated to determine the required beam weighting to approximate, in a minimum mean square error (MSE) sense, desired gain contours on the surface of the earth. One method is a functional minimization of the MSE between the actual and desired gain values using a gradient technique to search for the optimum weighting vector. A second method is the numerical solution of a system of simultaneous linear equations to find the beamweight vectors that minimize the MSE between the desired and actual gain contours. The latter approach was pursued first. An IMSL (International Mathematical and Statistical Library) library routine was selected to perform the numerical solution. The technique used by IMSL is the lower triangular-upper triangular (LU) decomposition (also named Cholesky reduction), a modification of the Gauss-Jordan elimination method more suitable for computer processing.

To test the algorithm, a singlet data matrix was generated using ideal singlet beam patterns based on an assumption of a uniformly illuminated circular aperture antenna. This matrix consists of 61 column vectors (corresponding to each of the 61 singlet beams) with the elements of each column vector corresponding to ideal element gain values at each point in the field of view. This matrix specifies the singlet beam patterns with 0.2° granularity over the full earth field of view of ±9° azimuth/elevation from the boresight of the center beam of the MBR. This results in 8281 points of interest (gain values) on the earth surface for each of the 61 singlet beams. The singlet data matrix is therefore of size 8281 x 61. As part of the initial test, a desired gain vector (8281 x 1) (i.e., full earth field of view) was selected with elements equal to one column of the singlet data matrix. That is, the desired gain contour was set equal to the gain contour of one singlet beam of the 61-beam MBR. The resultant weighting vector generated by the algorithm consisted of 60 zero weight elements and unity weighting for the beam whose singlet gain contour was equal to the desired gain contour. That is, to achieve the desired gain contour, the weights of all beams in the MBR were set to zero by the algorithm except the one whose gain matched that of the desired gain. This result provided a good initial test of the accuracy of the algorithm.

Further testing was performed using an actual contour pattern developed for continuous wave (CW) acceptance tests of the DSCS III A1. Data points for azimuth/elevation angle and gain level were obtained for the contours given by "sampling" every 0.2° of azimuth. A singlet data matrix was generated for these points using the ideal singlet beam approximation described above. The desired gain vector was obtained directly from the sampled contour data. A weighting vector was then computed and the resultant gain vector was determined. A contour plotting program was written and the contours (levels) of interest were drawn based on the resultant gain values. Simple desired contour patterns were approximated very closely. The more complex the desired contour, the more difficult it was to compute the appropriate beam weights to synthesize the desired pattern. This was a result of the use of ideal singlet data and the fact that the desired gain contours were obtained using actual (non-ideal) MBR singlet data. Once the actual MBR singlet data became available, a new beamweight vector was computed and the resultant gain vector was determined. The resultant contour plots using actual measured singlet beam patterns showed a much closer match to the desired contours than had previously been obtained using ideal singlet data.

Another test performed involved jammer nulling. The actual contour pattern described above was modified to include a jammer null at an arbitrary location. The beamweight vector was computed as before (using actual singlet data) for several different nulling depths at the jammer location of interest. The resultant gain distributions were determined and contours were plotted. Although excellent matches between desired and actual patterns were achieved, it was found that the greater the depth of the desired null relative to the close surrounding contours, the more difficult it was to match the desired contour patterns. This is due, in part, to the limited resolution of the 61 element earth coverage MBA and to the fact that only the "natural" phase of the MBA was used for the desired gain contour points instead of a phase requirement designed to emphasize the desired null.

The second algorithm (functional minimization) was developed in software and used to improve upon the results obtained using the first algorithm (LU decomposition). The functional minimization approach requires an initial guess for the beamweight vector which is modified to minimize the MSE between the actual and desired gain values. The initial beamweight vector used as input to the algorithm is obtained from the LU decomposition algorithm solution. A significant improvement in matching the resultant gain contours to the desired has been observed and is documented in section 3.6.1.

The model description, numerical methods and test results are described below.

3.3 MODEL DESCRIPTION

The MBR antenna is composed of 61 individual singlet beams. The maximum gain that each singlet beam can produce at a given azimuth/elevation (on the earth's surface relative to the satellite) is given by the singlet data. A singlet data matrix can be constructed for specific areas of interest on the earth surface. This matrix has the form:

$$A = [\alpha_{m,n} + j\ \beta_{m,n}]_{m,n} \qquad 3.1$$

where m = azimuth/elevation coordinate index; $m = 1,2, \ldots, M$

n = singlet beam index; $n = 1,2, \ldots, N$

That is, each of the M rows of A corresponds to a specific azimuth/elevation

coordinate point within the area of interest and each of the N columns corresponds to a particular singlet beam. The maximum coordinate index M may vary from one point to full coverage of 8281 points[1]. The number of columns of A will generally equal 61, since all 61 beams normally would be considered. Each A matrix element is a complex number representing the real and imaginary components of the maximum gain of a given singlet beam at an azimuth/elevation point. The MBR singlet beams are electronically weighted and summed to produce a desired composite gain distribution on the earth surface. Algebraically, this is equivalent to multiplying the singlet data matrix A times a beamweight column vector $\underline{W}$ consisting of 61 complex elements, each element corresponding to the weighting applied to an associated singlet beam. The result is a gain vector $\underline{G}$ whose elements represent, in general, the complex gain at each azimuth/elevation point within the areas of interest defined by the A matrix. Symbolically,

$$\underline{G} = A\underline{W} \qquad 3.2$$

where $A = [\alpha_{m,n} + j\, \beta_{m,n}]_{m,n}$

$\underline{W} = [\gamma_n + j\, \delta_n]_{n,1}$

$\underline{G} = [\lambda_m + j\, \mu_m]_{m,1}$

3.4 NUMERICAL ANALYSIS TECHNIQUES

3.4.1 Algorithm 1: Minimum Mean Square Error Solution for Complex Gain Contours

The MBR antenna beam forming problem is to compute the beam weights for the 61-element antenna to synthesize a given desired gain contour. Two approaches were considered. The first approach is to specify the desired gain contour in terms of magnitude and phase (complex) and to minimize the weighted MSE given by

$$\varepsilon^2 = (\underline{G}_a - \underline{G}_d)^* B (\underline{G}_a - \underline{G}_d) \qquad 3.3$$

where $\underline{G}_d$ = desired gain vector (M x 1)

$\underline{G}_a$ = actual gain vector (M x 1)

B = diagonal weighting matrix related to the relative importance of areas of interest (for unbiased weighting of all points within the coverage area, B is an identity matrix)

---

[1] Full earth coverage consists of ± 9° azimuth and elevation, with 0.2° granularity, from the subsatellite point of a geosynchronous satellite.

$(\bullet)^*$ = complex conjugate transpose of a given matrix or vector

Since the actual gain vector $\underline{G}_a$ may be written as

$$\underline{G}_a = A\underline{W} \quad 3.4$$

where A = singlet data matrix (M x 61)
$\underline{W}$ = beamweight vector (61 x 1)

the weighted mean-square-error function ε^2 becomes

$$\varepsilon^2 = (A\underline{W} - \underline{G}_d)^* B (A\underline{W} - G_d) \quad 3.5$$

Setting the gradient of ε^2 equal to zero yields a system of 61 equations in 61 unknowns given by

$$A^*BA\underline{W} = A^*B\underline{G}_d \quad 3.6$$

Many standard numeric analysis methods can be used to solve this system for the beam weight vector $\underline{W}$. An IMSL FORTRAN library routine was selected to perform the numerical solution. The technique used by IMSL is an LU decomposition, a modification of the Gauss-Jordan elimination method, more suitable for computer applications. The matrix of coefficients A^*BA is transformed into the product of two matrices L and U, where L is a lower triangular and U is an upper triangular matrix with 1s on its diagonal. The advantage is that storage space is economized and the ability to solve the system of equations with new righthand-side vectors $A^*B\underline{G}_d$ (i.e., the required gain contours G_d) can be achieved with great economy of effort. The number of arithmetic operations to obtain the solution corresponding to each $A^*B\underline{G}_d$ is the same as to multiply an N x M matrix by an M-component vector. Since both magnitude and phase of the gain are specified with this method, it is useful when phase tapering (i.e., adding a fixed phase offset between singlet beams) is desired.

3.4.2 Algorithm 2: Minimum Mean Square Error Solution for Magnitude-Only Gain Contours

A second approach to computing the beam weight vector is to specify the gain by magnitude only and to minimize the weighted mean-square-error given by

$$|\varepsilon^2| = (|\underline{G}_a| - |\underline{G}_d|)^T B (|\underline{G}_a| - |\underline{G}_d|) \qquad 3.7$$

$$= \sum_{i=1}^{m} B_{ii}(|\underline{G}_{a_i}| - |\underline{G}_{d_i}|)^2$$

A functional minimization of $|\varepsilon^2|$ using the Davidon-Fletcher-Powell (DFP) algorithm can now be employed. This method is superior to the first if only the magnitude of the gain is desired. Since the phase at each point of the desired gain contour is unspecified with this method, an additional degree of freedom exists over the first method for determining a solution. Of course, the phase of each component of the desired weight vector will still be of critical importance in establishing any desired pattern nulls. However, due to the gradient search technique employed with this method, computational time may be greater than that of algorithm 1.

3.5 VERIFICATION OF ALGORITHM 1

Two tests were performed using the first algorithm described in order to determine the accuracy of the solution and to verify that the software is correct. As discussed in Section 3.1, this algorithm minimizes the weighted mean square error when the desired gain values are specified as complex numbers (magnitude and phase). For these tests, it was assumed that all gain values are of equal importance. Therefore, the weighting matrix B was set equal to the identity matrix.

3.5.1 Initial Test

An approximation to the beam pattern of each singlet beam was based on that of a uniformly-illuminated circular aperture. The directivity pattern is given by:

$$E(u) = 2\pi a^2 J_1(u)/u \qquad 3.8$$

where a = radius of the circular aperture

u = $(2\pi a/\lambda)\sin\Theta$

λ = transmission wavelength

Θ = angle from antenna boresight (the look angle)

$J_1(.)$ = first-order Bessel function

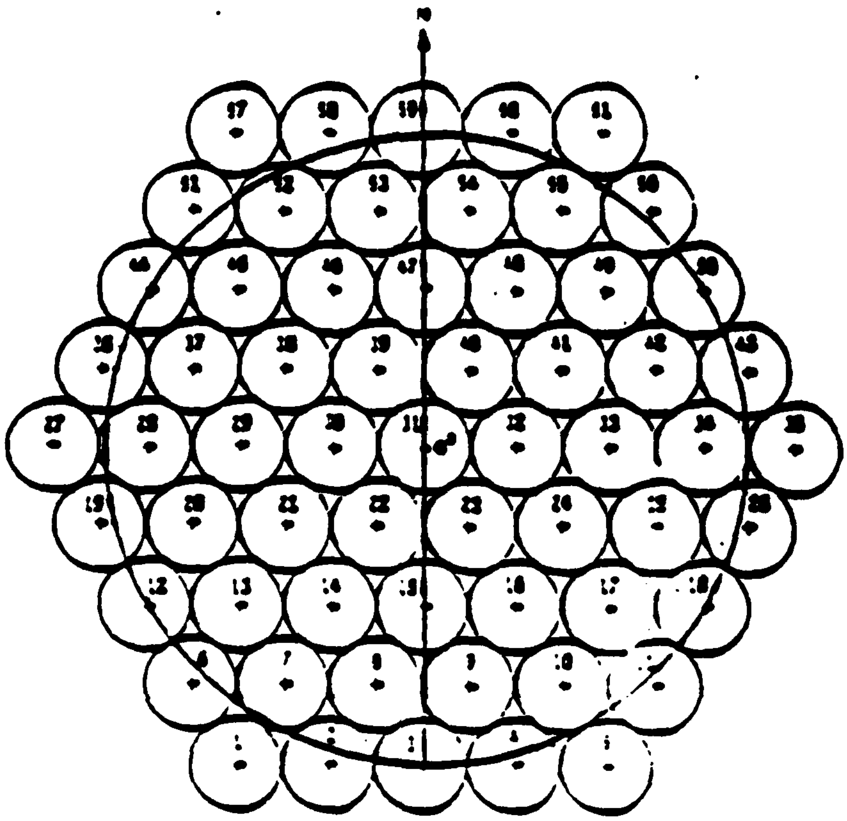


Figure 3-1. Boresight Pattern of 61-Beam MBR Antenna Superimposed on Earth

To determine the gain from each of the 61 beams in the MBR, the angle Θ was computed as the angular distance from singlet beam boresight coordinate to the azimuth/elevation coordinate of interest. Figure 3-1 shows the boresight pattern of the 61-beam MBR antenna superimposed on the earth. The concentric circles represent the half power (3 dB) beamwidth of each singlet beam. Since the DSCS III MBR is 1.17 meters in diameter and assuming the transmission wavelength λ to be 0.0375 meters corresponding to a transmission frequency of 8 GHz, the 3-dB beamwidth of a singlet beam can be computed. The result is a beamwidth of approximately 2.25° diameter. The

azimuth/elevation coordinate of each boresight now can be computed using the center beam boresight located at 0° azimuth/0° elevation as reference. The magnitude of the singlet gain for the center beam of the MBR (beam #31) vs azimuth angle at a fixed elevation angle of 0 degrees is shown in Figure 3-2.

A singlet data matrix was generated, using the above approximation (an IMSL library routine was used to compute the Bessel function), consisting of the full earth coverage area of $\pm 9°$ azimuth/elevation (0.2° granularity) from the boresight of the center beam of the MBR. This is equivalent to superimposing a $\pm 9°$ azimuth/elevation grid on the boresight pattern of Figure 3-1. The result is 8281 points of interest (gain values) on the earth surface for each of the 61 singlet beams. The singlet data matrix is therefore of size 8281 x 61.

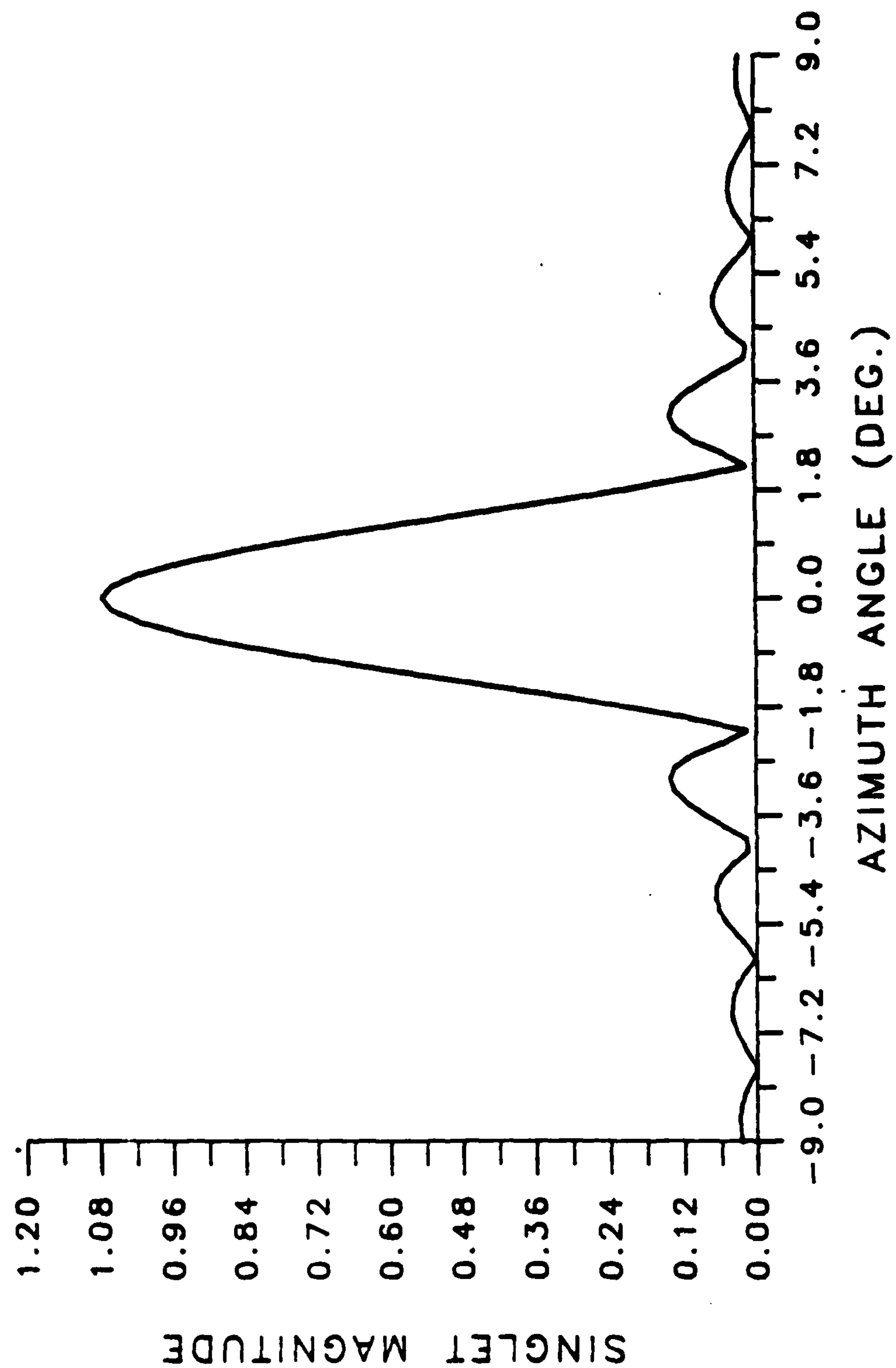
3.5.1.1 Results of Initial Test

A desired gain vector of size 8281 x 1 was generated whose elements were equal to one column of the singlet data matrix. That is, the desired gain was set equal to that produced by a single active beam. The computed beam weight vector consisted of one non-zero element with a weighting of one for the beam whose singlet gain was equal to the desired gain. That is, to achieve the desired gain, the algorithm disabled all beams in the MBR except for a unity weight on the beam whose gain matched that of the desired gain. This was the expected result and indicates the algorithm to be correctly implemented.

3.5.1.2 Computation Time Required

The time required for solution can be considered in two parts. The first is the time required to generated the system of equations. This involves multiplication of A^*BA where A in this case is a full size 8281 x 61 singlet data matrix. The total required CPU time was 3 hours. The second is the time required to solve the system of simultaneous equations for the $\underline{W}$ vector. The total CPU time for this was 2 seconds. For a given singlet data matrix A, which is developed for a defined set of contour points on the earth's surface, the multiplication A^*BA needs to be performed only once, even if the gain values for each of the defined contours are changed (unless the importance of particular points in the gain contour change, requiring a change in the B matrix). That is, if the gains $\underline{G}_d$ of a given contour are changed in successive cases, the singlet matrix A remains unchanged and successive beam weight vectors $\underline{W}$ can be computed, each

Figure 3-2. Singlet Gain of Beam #31 at Elevation=0.0 for a Uniformly-Illuminated Circular Aperture



solution requiring approximately 2 seconds of CPU time. Most gain contours would have many fewer points than 8281 (the maximum) and the computation of A^*BA typically would be much less than 3 hours of CPU time. Furthermore, for cases where the relative importance of given user locations is changed (i.e., requiring a change to the weighting matrix B), it is advantageous to the user that gains be specified only for those points that are critical to the mission in order to reduce the A^*BA computation time. Of course, for cases where the exact user location is unknown or specified as an operating area of interest, a sufficient number of points within the operating area must be included within the required gain contour. In practice, for a given set of contour points, A^*BA could be stored offline on disk and called up when needed.

3.5.2 Test with Actual Gain Contours

To test Algorithm 1 with realistic desired gain contours, an actual contour pattern developed for CW acceptance tests of the DSCS III A1 was used. This contour is shown in Figure 3-3. Input data points for azimuth/elevation angle and gain level were obtained for the contours given by "sampling" every 0.2° of azimuth ($M = 900$). The sampled contour data are shown in Figure 3-4.

The desired gain vector was obtained directly from the sampled contour data.

3.5.2.1 Results Using Ideal Singlet Data

A singlet data matrix was generated for the points of interest (contour points) using the Bessel function approximation described previously. Only the magnitude of the singlet data is computed from this approximation. The phase is assumed to be zero. Since only the magnitude of the desired gain is specified by the contour data, the phase of the gain at each point can be considered arbitrary. Since the phase associated with the singlet data approximation was assumed to be zero, the phase of the gain was also set to zero. A beam weight vector was computed with the above test data using algorithm 1. The resultant contour gain values for the points of interest were then computed by premultiplying the computed beam weight vector by the singlet data matrix. A line graph of the resultant gain values for each desired contour level versus data point is shown in Figure 3-5. The resultant gain values for desired 16-dB points show the least variation of the three levels of interest (approximately ± 1 dB). The resultant gain values for the 11-dB desired points exhibit the largest variation.

Figure 3-3. Contour Patterns for MBA used in CW Tests

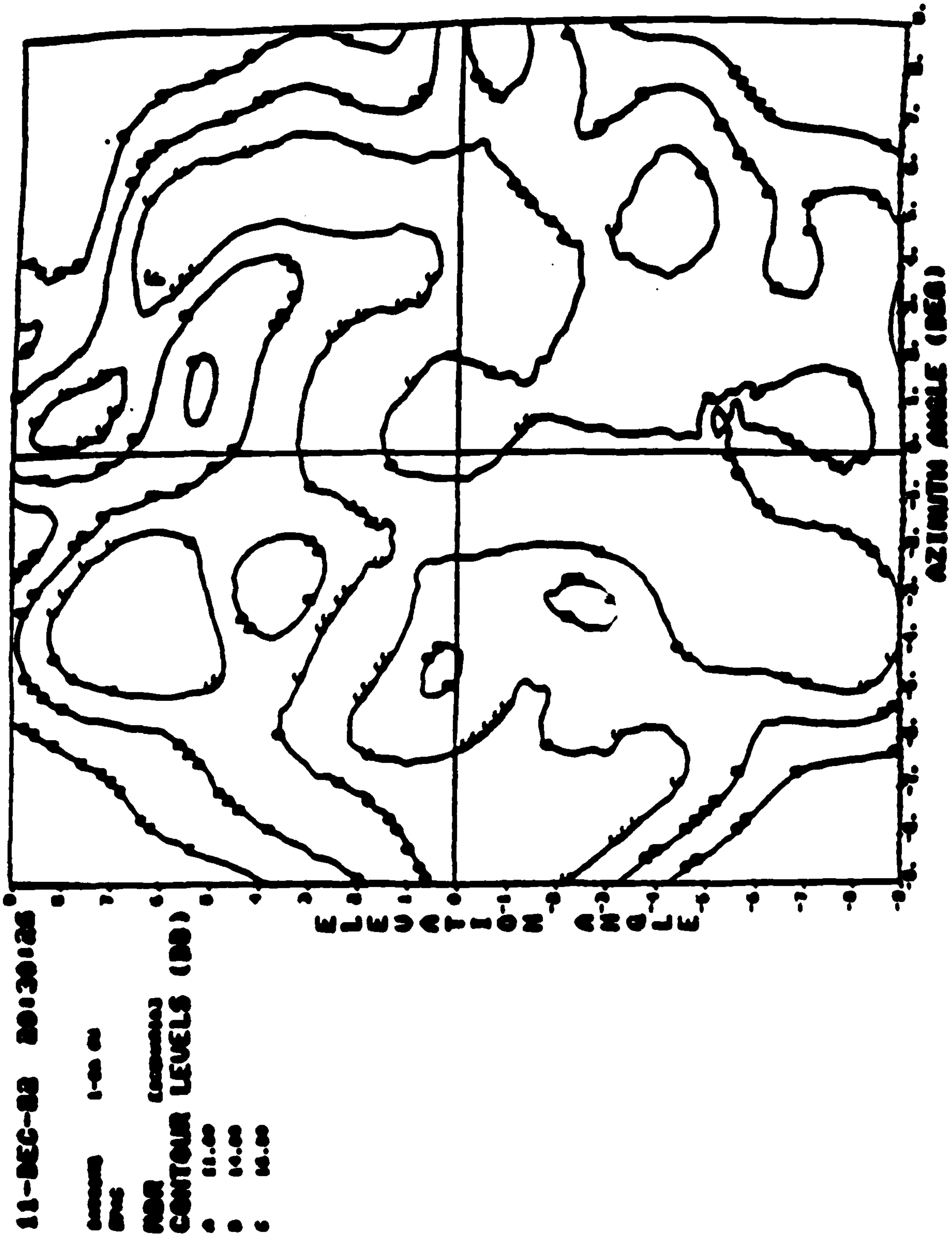


Figure 3-4. Sampled Contour Data

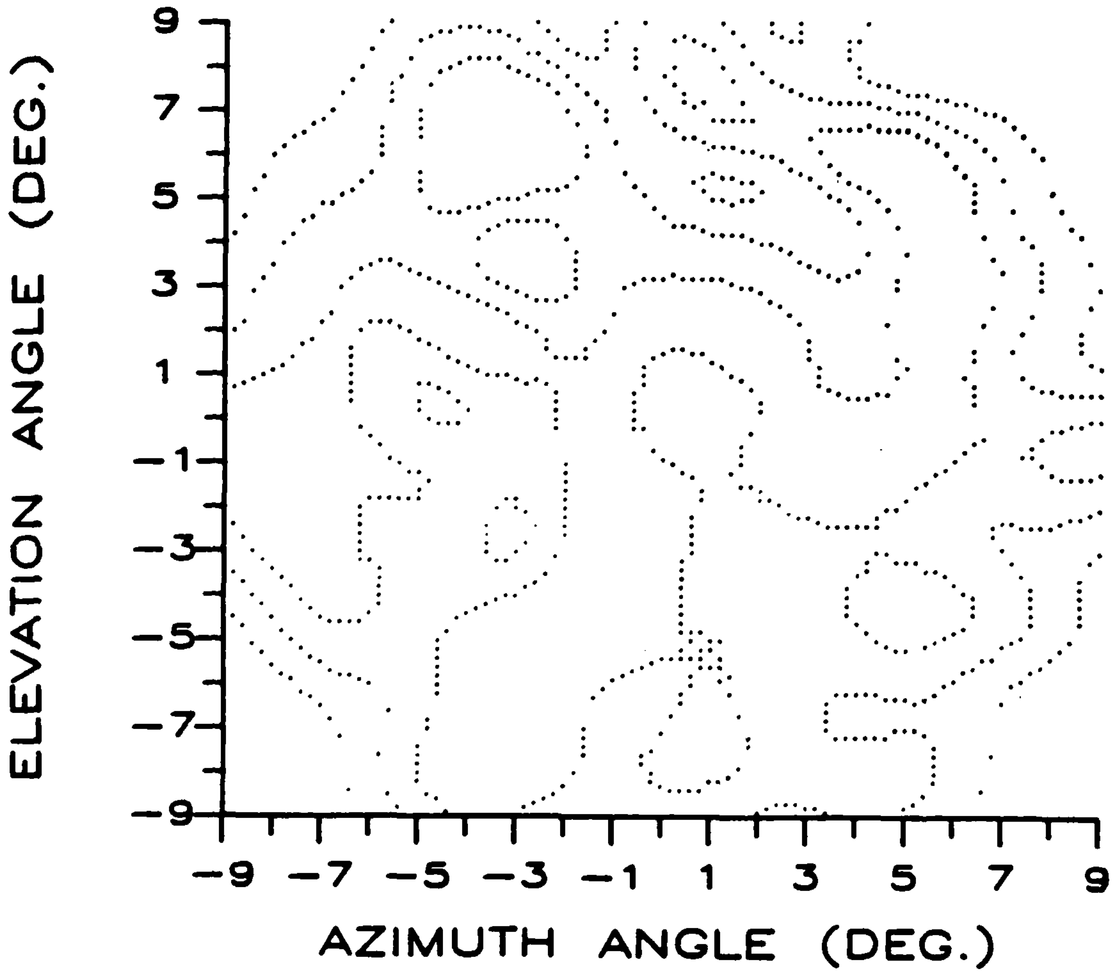
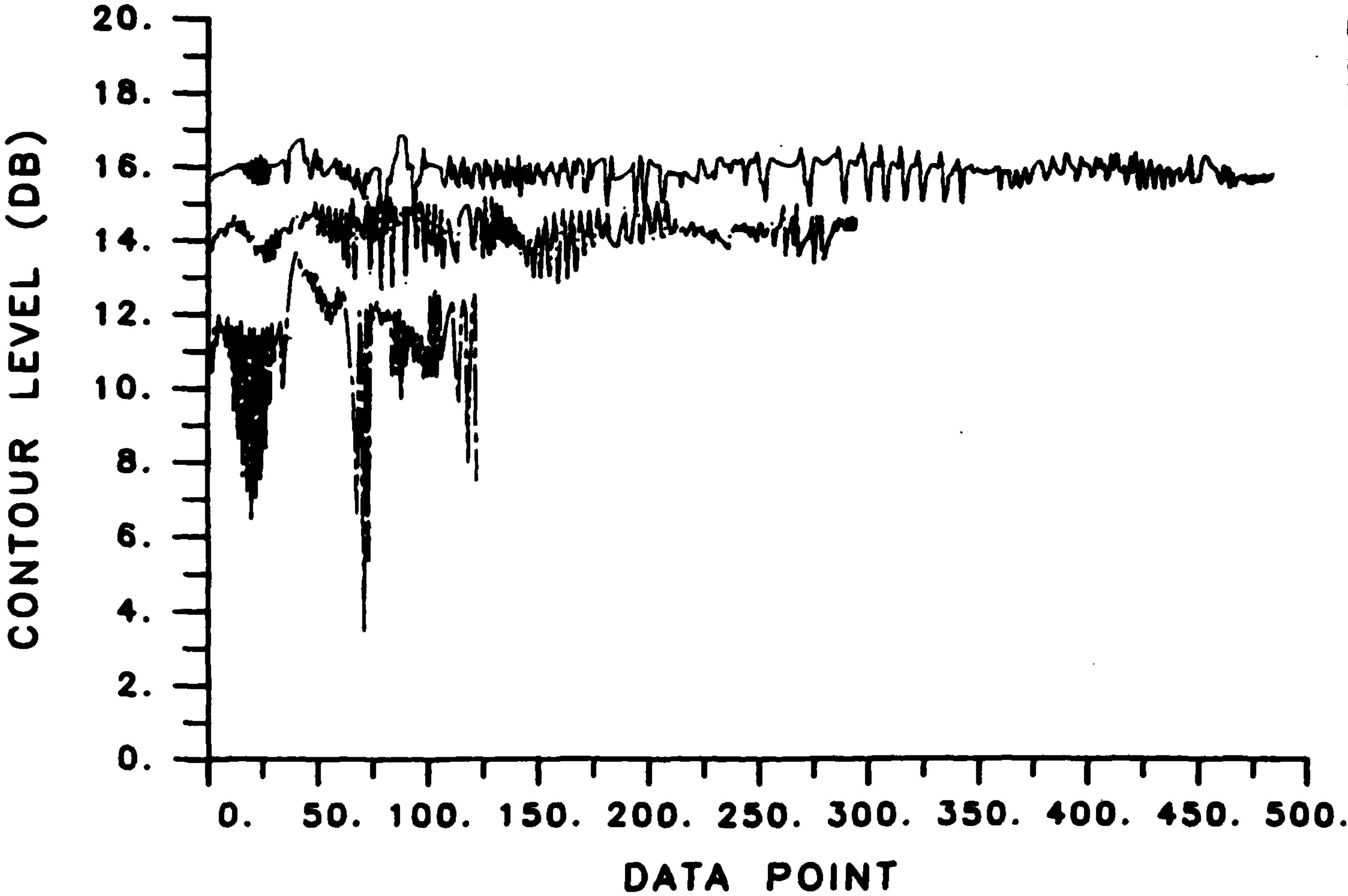


Figure 3-5. Resultant Gain vs. Data Point - Ideal Singlet Data



To compare the desired and resultant contour patterns it was necessary to compute the resultant gain distribution over the full ±9° azimuth/elevation grid and plot the contour levels of interest. The resultant gain distribution was computed by premultiplying the beam weight vector by the full 8281 x 61 singlet data matrix.

A simple contour plotting program was written to generate the resultant contour levels of interest. This program is designed to scan the ±9° azimuth/elevation grid in both azimuth and elevation and to mark points where transitions in a gain level occur across levels of interest. The result is a series of points indicating transitions that can be connected to form gain contours.

The desired 16-dB gain contours, obtained directly from the sampled data, are shown in Figure 3-6. The resultant 16 dB contour computed from the algorithmically determined beamweighting vector is shown in Figure 3-7. When the resultant contour is overlaid on the desired one, it can be seen that certain areas are matched quite closely while others are not. Due to the complexity of the 16-dB contour and the fact that an approximation to the singlet data was used, an exact match between desired and actual contours was not expected. Similarly, the desired and resultant 14-dB contours are shown in Figures 3-8 and 3-9, respectively. The 14-dB contour is much simpler than the 16-dB and a closer match is seen between the desired and actual patterns. Finally, the desired and resultant 11-dB contours are shown in Figures 3-10 and 3-11, respectively. This is the simplest contour and shows the closest match between the desired and actual patterns.

3.5.2.2 Results Using Actual Singlet Data

The actual singlet data were obtained and a new beam weight vector was computed. The actual singlet data comprise a complex vector (i.e., in-phase and quadrature) with components that represent the magnitude and phase of the MBR gain per beam at each point on the earth with unity beamweights. The normalized magnitude of the gain for the center beam of the MBR (beam #31) vs azimuth angle at an elevation of 0 degrees is shown in Figure 3-12. This can be compared with Figure 3-2 which represents an approximation to the actual singlet data. Figure 3-13 shows the corresponding phase response confined to the range $-\pi$ to π (i.e., plotted modulo 2π). For comparison, the singlet gain and phase of the edge beams at an elevation angle of 0 degrees (i.e., beams 27 and 35) are shown in Figures 3-14 thru 3-17. Of interest is the difference in phase

Figure 3-6. Desired 16 dB Contour

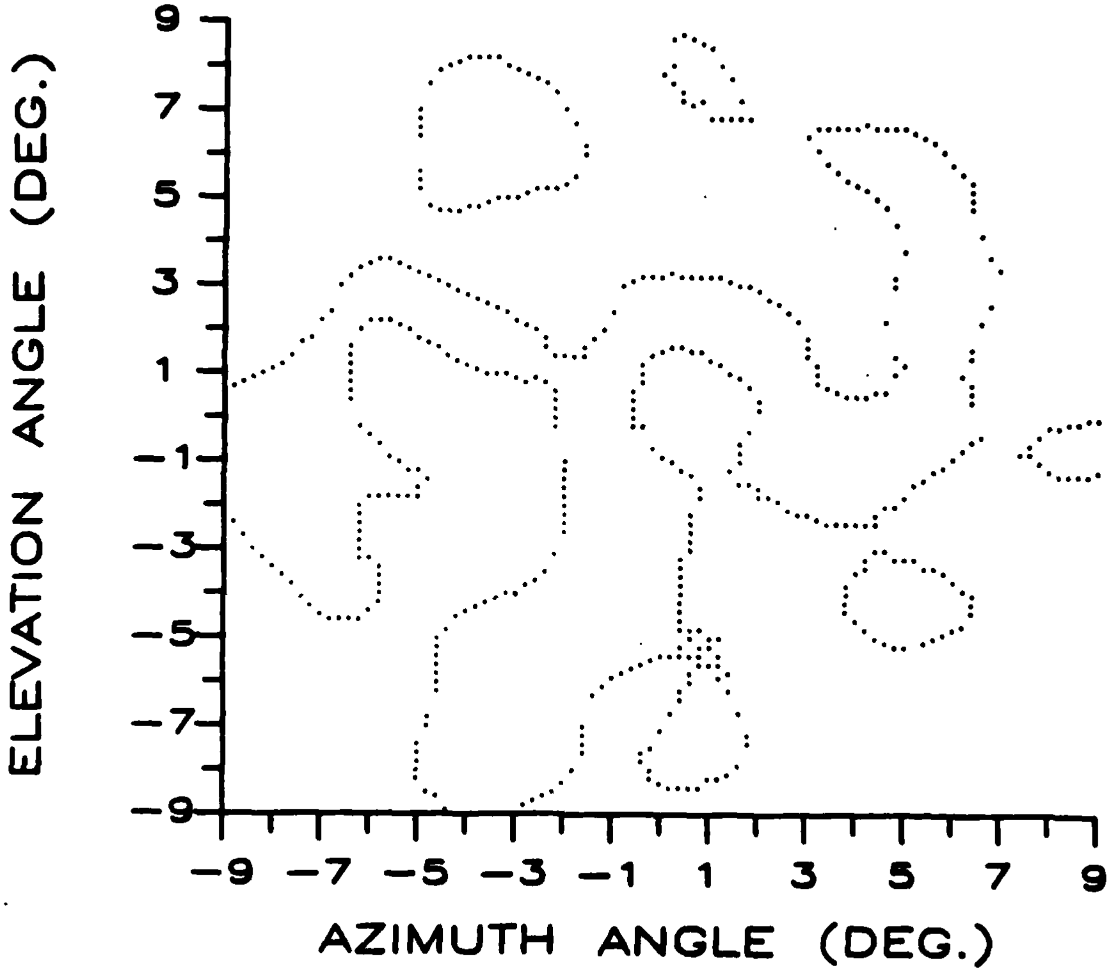


Figure 3-7. Resultant 16 dB Contour - Ideal Singlet Data

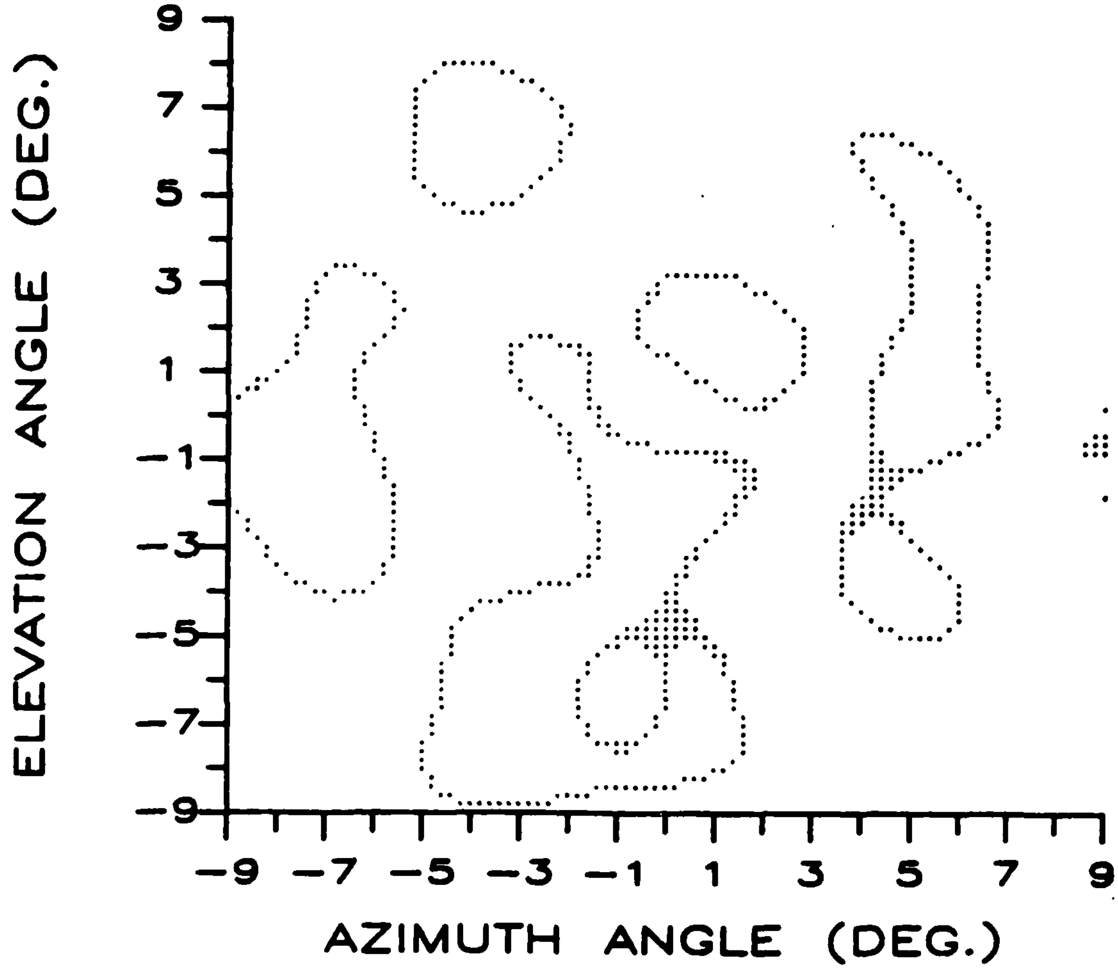


Figure 3-8. Desired 14 dB Contour

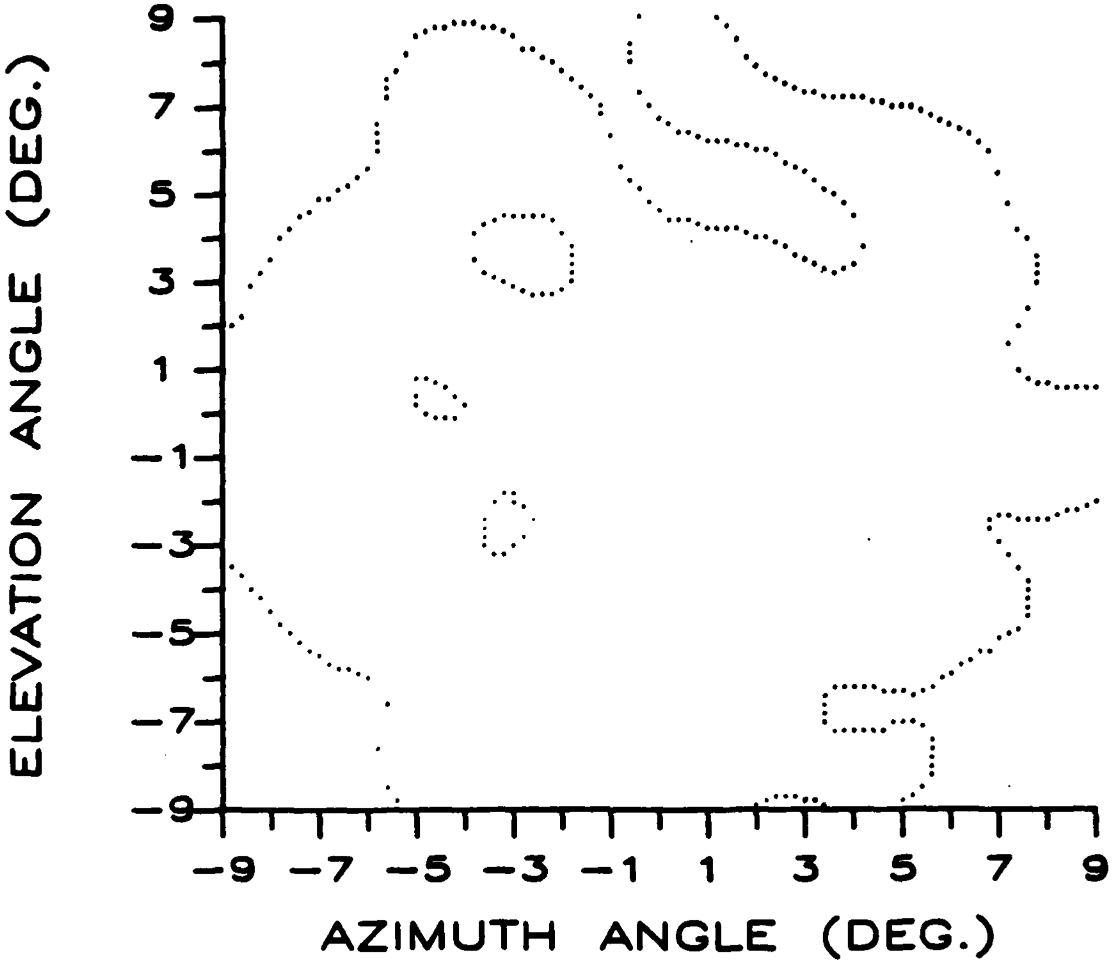


Figure 3-9. Resultant 14 dB Contour - Ideal Singlet Data

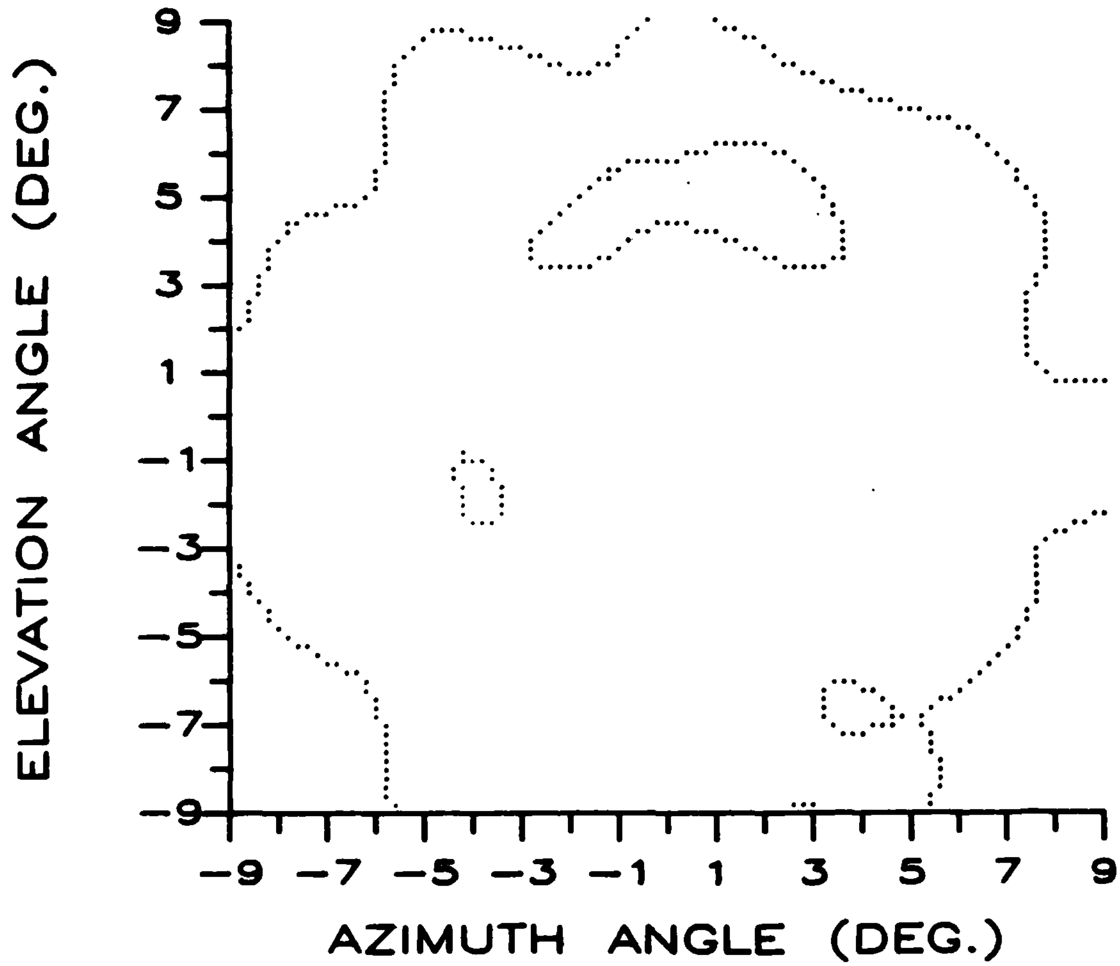


Figure 3-10. Desired 11 dB Contour

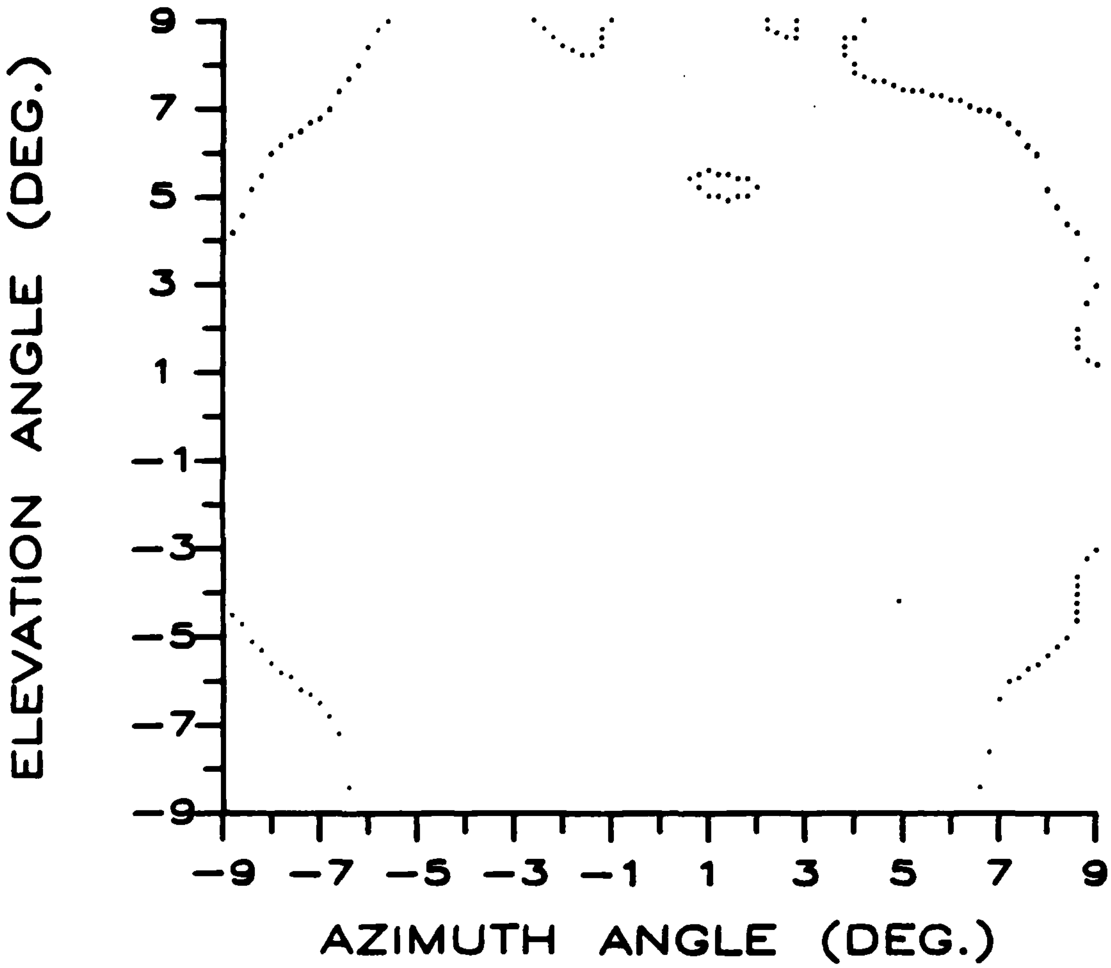
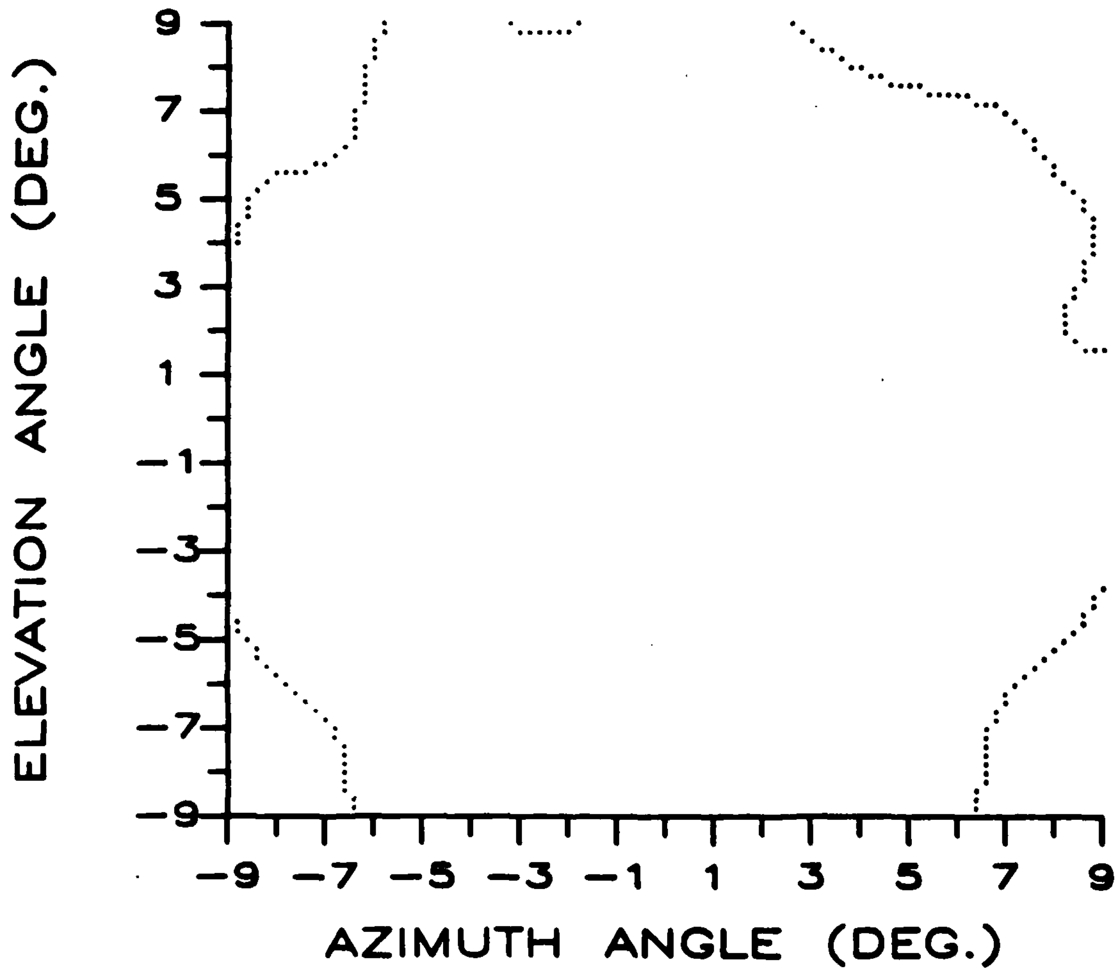


Figure 3-11. Resultant 11 dB Contour - Ideal Singlet Data



angle across the main gain lobe between the three beams shown. This phase angle difference is due to the fact that all beam feed horns are not at the MBA reflector focal point.

Since no phase information was associated with the desired gain contours, the phase of the desired gain at each point of interest was assigned a value equal to the phase of the vector sum of the singlet data for the 61 beams at the point of interest on the earth's surface, i.e., the natural phase response of the aperture at all points of interest on the earth's surface. In other words, the phase assigned to a desired gain point M is the phase of the complex sum of row M of the singlet data matrix. In equation form this is given by:

$$\theta_M = \tan^{-1}\left[\sum_{i=1}^{61} \beta_{M_i} \Big/ \sum_{i=1}^{61} \alpha_{M_i}\right] \qquad 3.9$$

where, α_{M_i} = real part of singlet data for the i^{th} beam at point M

β_{M_i} = imaginary part of singlet data for the i^{th} beam at point M

The required beamweight vector was computed using algorithm 1, and the resultant contour gain values for the points of interest were computed using Equation 3.4 where A represents the actual singlet data matrix. The beamweight vector W (without scaling of singlet data) is given in Table 3-1. A line graph of the resultant gain values for each desired contour level versus data point is shown in Figure 3-18. Comparison of Figure 3-18 (obtained with the actual singlet data) and Figure 3-5 (obtained with idealized singlet data) indicates a much improved capability to provide a contour match.

As in the previous section, to compare the desired and resultant contour patterns it is necessary to compute the resultant gain distribution over the full $\pm 9°$ azimuth/elevation grid and plot the contour levels of interest. The resultant gain distribution is computed by premultiplying the beam weight vector W by the full 8281 x 61 singlet data matrix.

The resultant 16 dB contour computed from the new beam weight vector (using the actual singlet data) is shown in Figure 3-19. The 14 dB and 11 dB contours are shown in Figures 3-20 and 3-21 respectively. The resultant contours are much closer to the corresponding desired contours when using the actual singlet data than those obtained

Figure 3-12. Singlet Gain of Beam #31 at Elevation=0.0

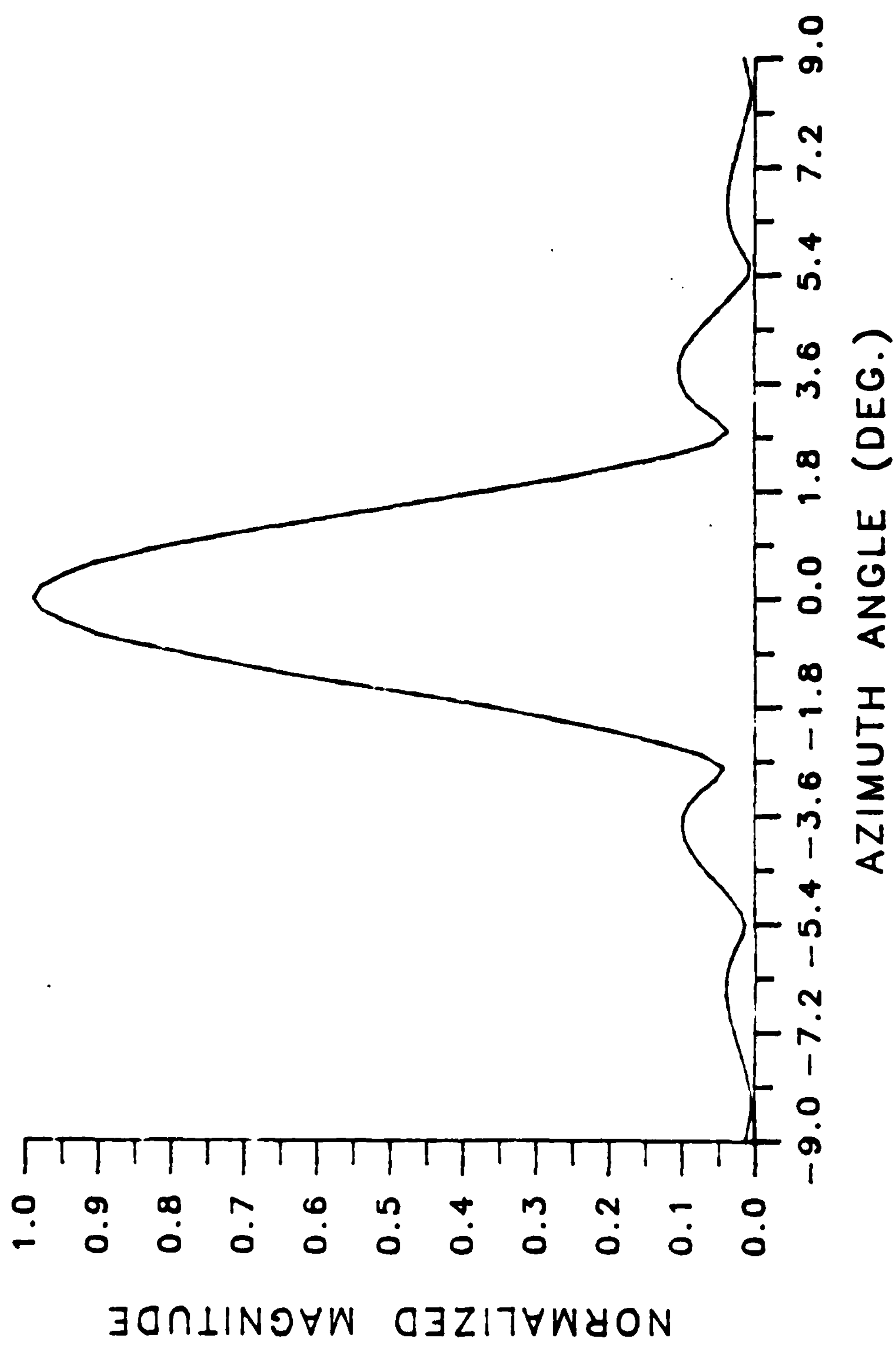


Figure 3-13. Singlet Phase of Beam #31 at Elevation=0.0

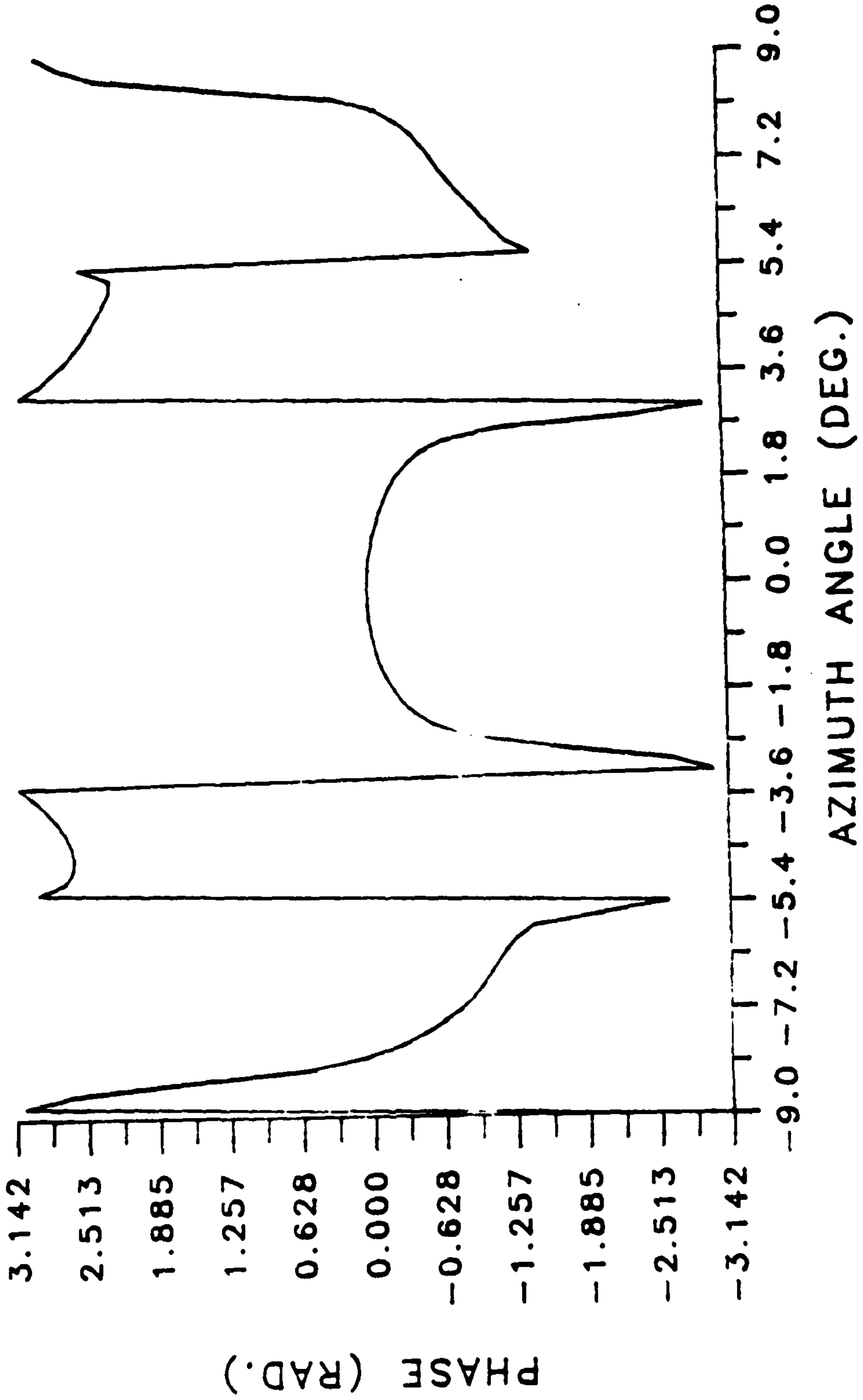


Figure 3-14. Singlet Gain of Beam #27 at Elevation=0.0

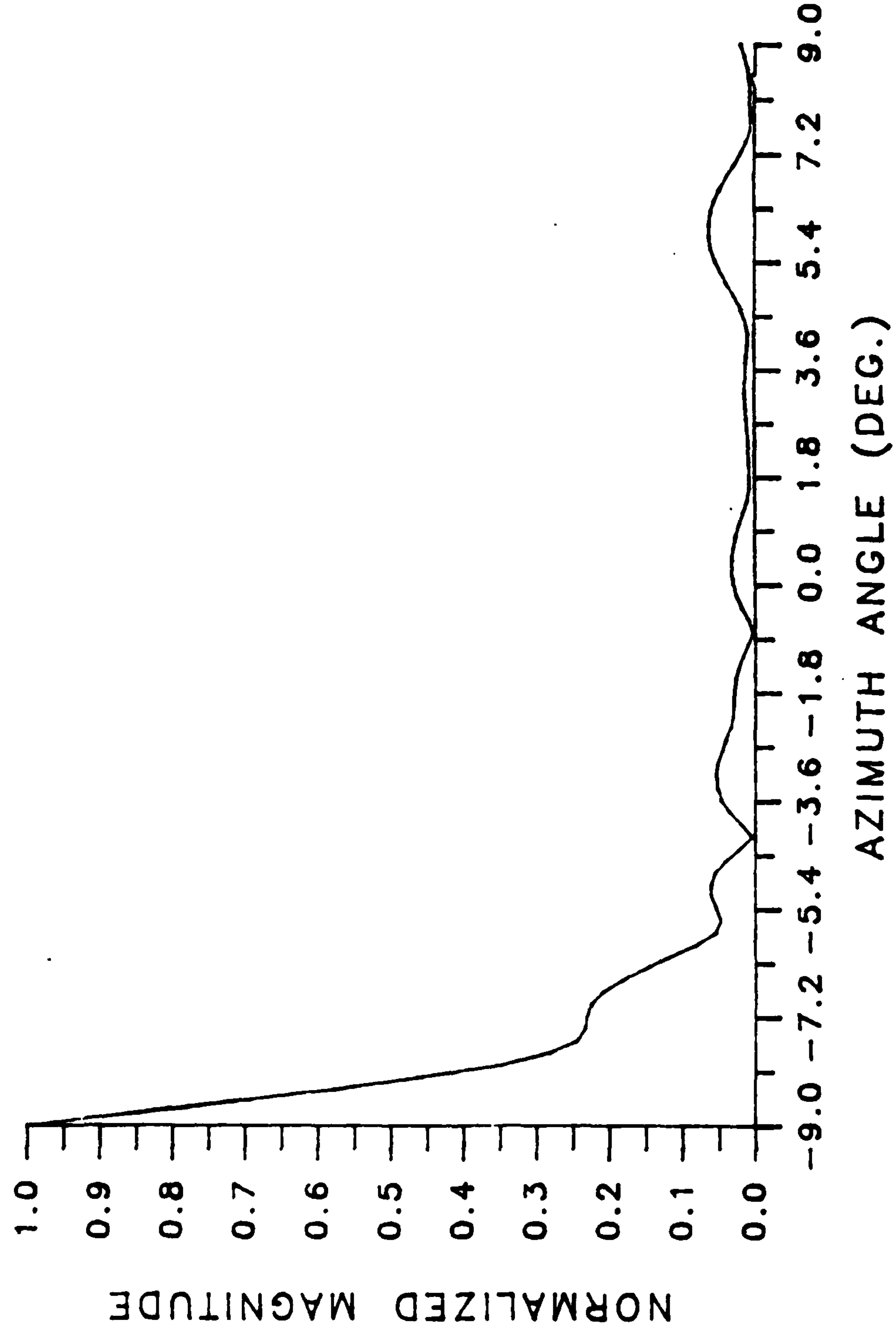


Figure 3-15. Singlet Phase of Beam #27 at Elevation=0.0

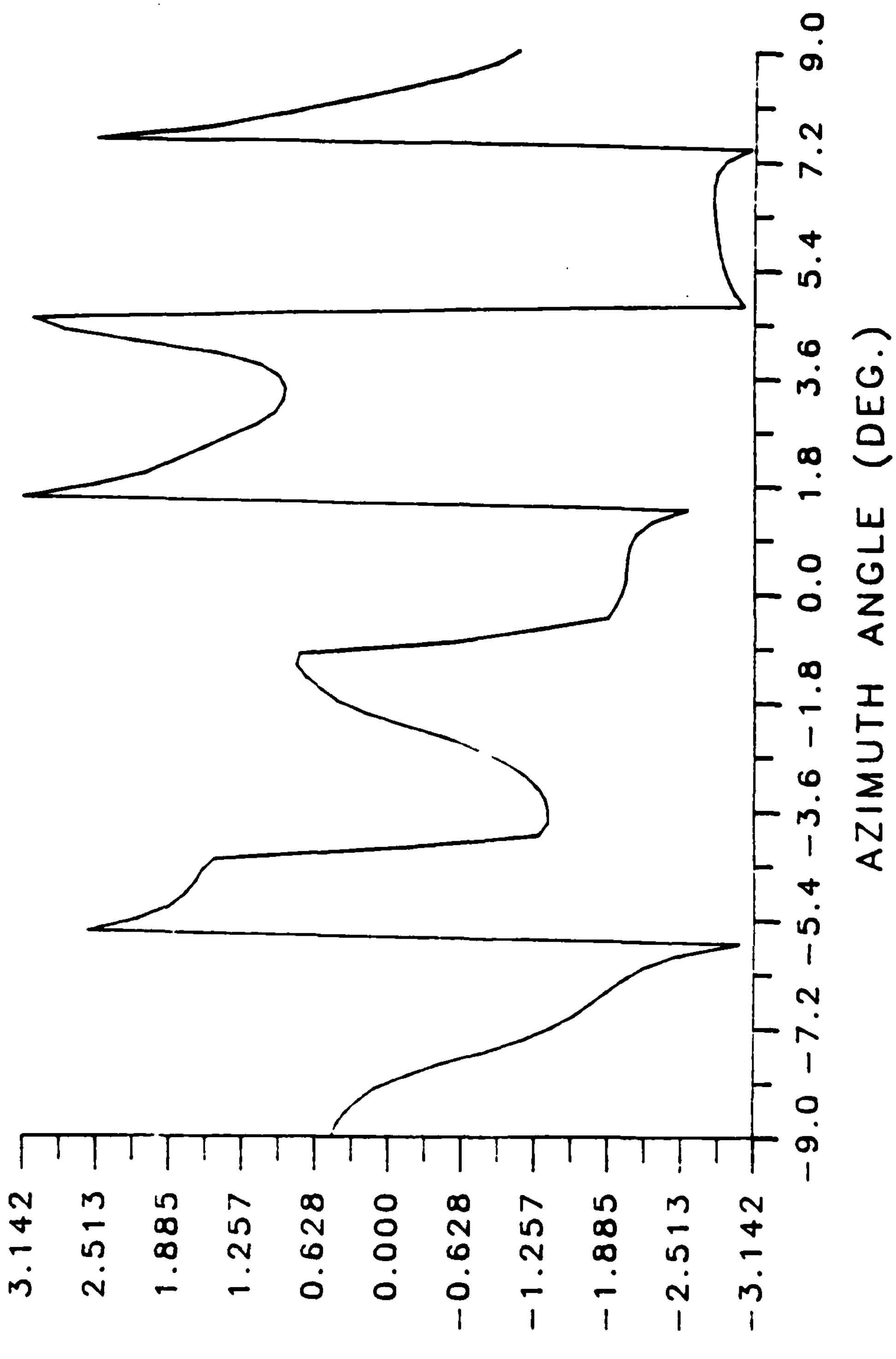


Figure 3-18. Singlet Gain of Beam #35 at Elevation=0.0

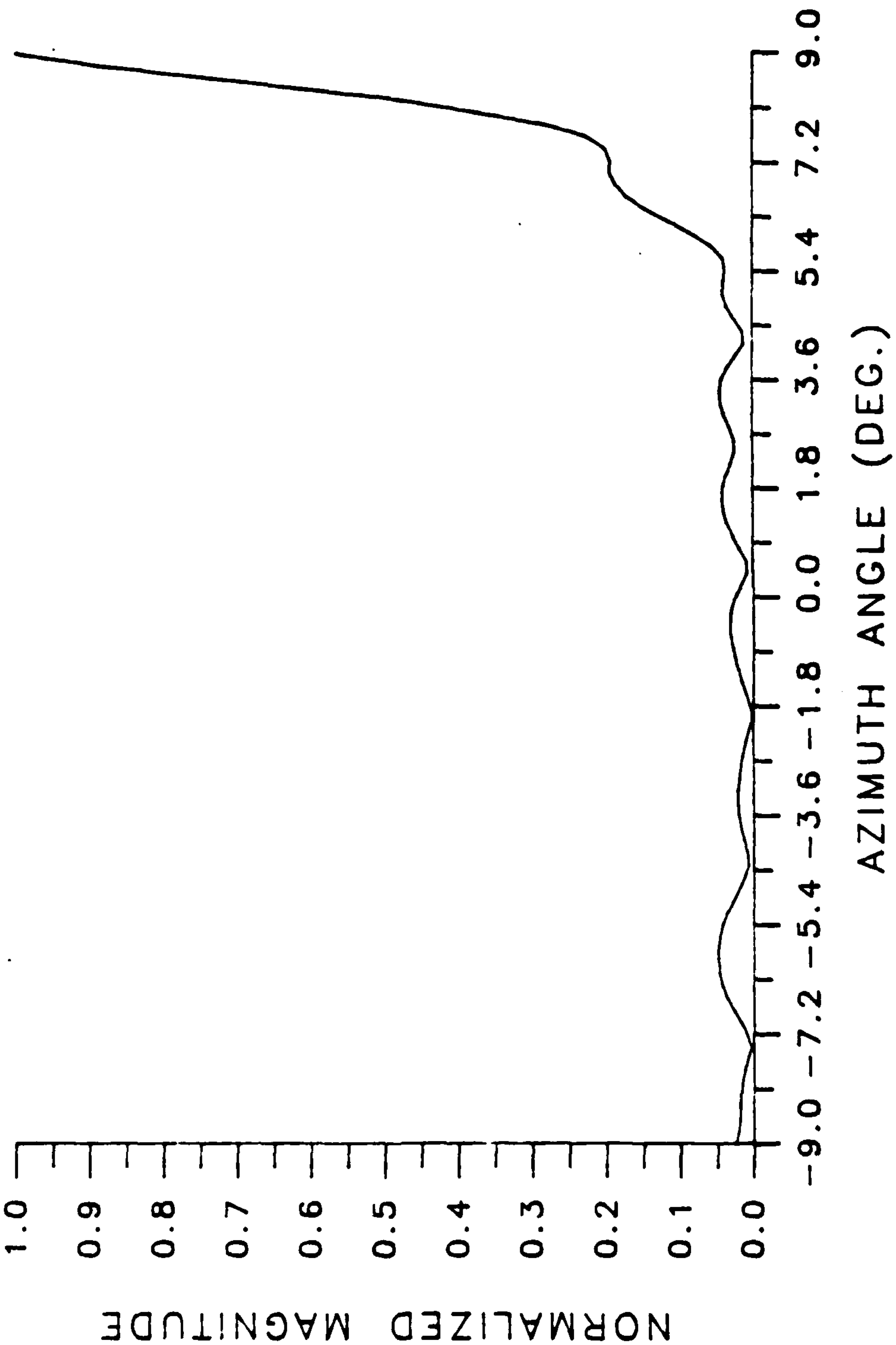


Figure 3-17. Singlet Phase of Beam #35 at Elevation=0.0

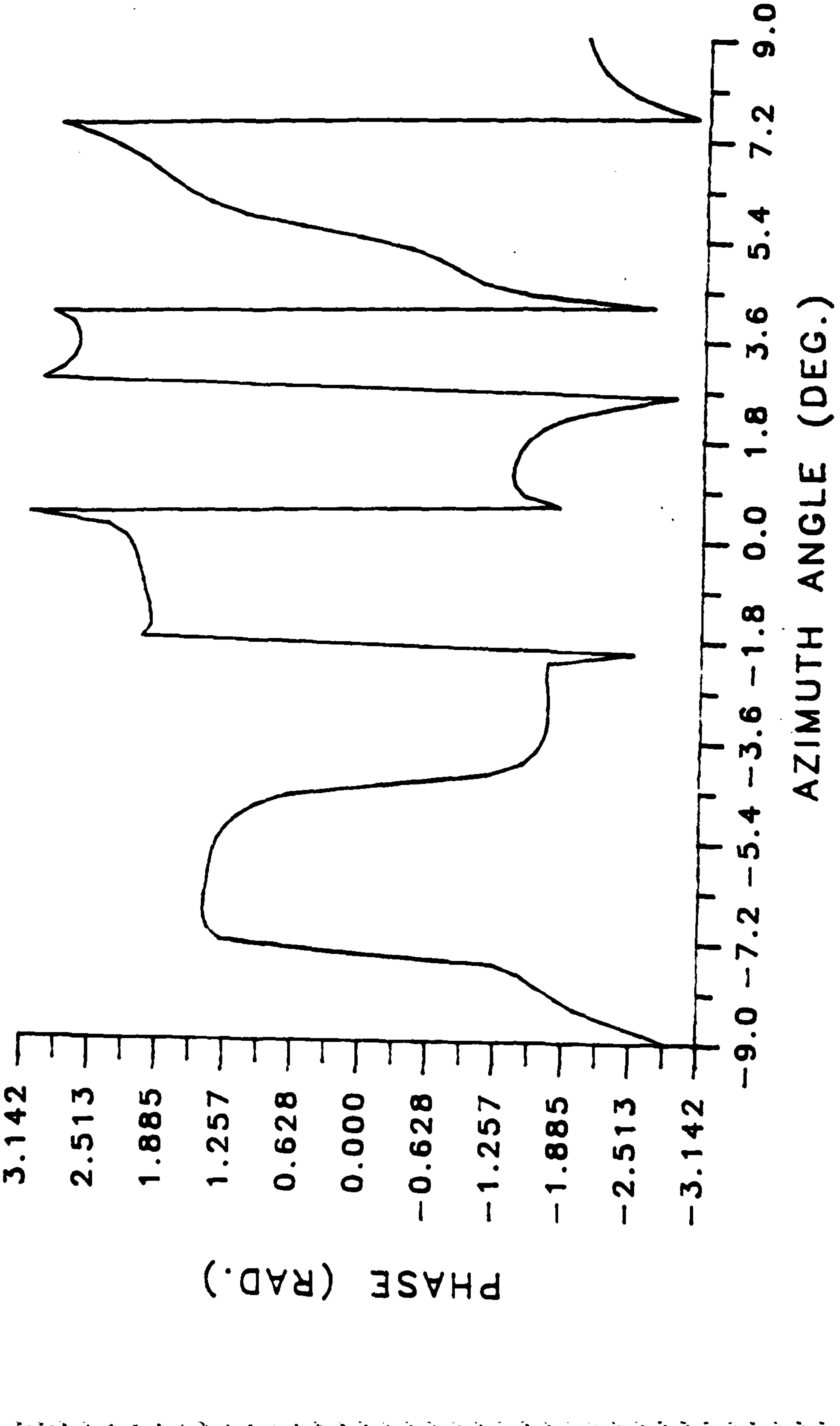
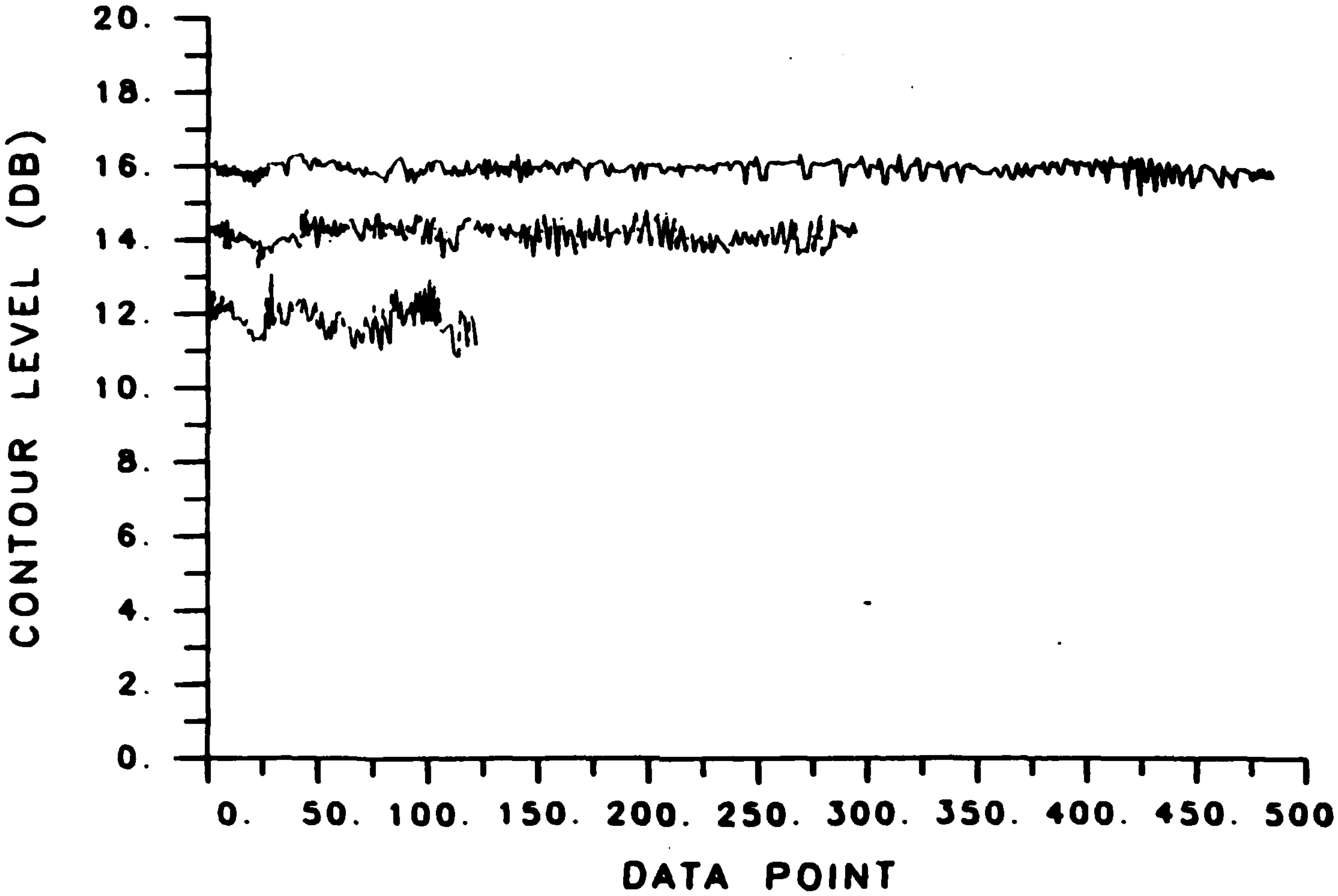


Table 3-1. Computed Beamweight Vector for MBR Contour Pattern used in CW Tests – no scaling

BEAM NO.	REAL	IMAG.	BEAM NO.	REAL	IMAG.
1	0.0009860	0.0000213	50	0.0009087	0.0000237
2	0.0010225	-0.0000157	51	0.0008072	0.0001820
3	0.0011401	0.0000170	52	0.0014895	-0.0000539
4	0.0008570	0.0000529	53	0.0009990	0.0000242
5	0.0009027	-0.0000442	54	0.0010151	0.0000250
6	0.0008050	0.0001552	55	0.0011112	-0.0000395
7	0.0011919	0.0001128	56	0.0008681	0.0001346
8	0.0010420	0.0000685	57	0.0008516	0.0000924
9	0.0011030	-0.0000786	58	0.0008251	0.0002572
10	0.0008444	0.0000667	59	0.0009666	-0.0000595
11	0.0008192	0.0001517	60	0.0007636	0.0001058
12	0.0010067	0.0000499	61	0.0007855	0.0004017
13	0.0011103	-0.0000284			
14	0.0010742	0.0000169			
15	0.0011252	-0.0000277			
16	0.0009145	0.0001107			
17	0.0012287	-0.0001138			
18	0.0006936	0.0000405			
19	0.0010486	0.0000109			
20	0.0011637	-0.0001128			
21	0.0009240	0.0001289			
22	0.0011397	0.0000333			
23	0.0010846	-0.0000394			
24	0.0010905	0.0000255			
25	0.0009402	0.0000118			
26	0.0008754	-0.0000242			
27	0.0012436	0.0000583			
28	0.0011944	-0.0000604			
29	0.0009949	0.0000832			
30	0.0010289	-0.0000555			
31	0.0010878	0.0000525			
32	0.0010371	-0.0000462			
33	0.0011702	0.0000122			
34	0.0010346	-0.0000051			
35	0.0011323	0.0000431			
36	0.0009555	0.0001012			
37	0.0011110	-0.0000553			
38	0.0010650	0.0000085			
39	0.0010713	-0.0000033			
40	0.0012437	-0.0000405			
41	0.0010192	0.0000443			
42	0.0010721	-0.0000286			
43	0.0007444	0.0003133			
44	0.0007516	-0.0000019			
45	0.0010580	0.0000765			
46	0.0008805	0.0001184			
47	0.0008926	0.0000659			
48	0.0008499	0.0002212			
49	0.0011781	-0.0000602			

Figure 3-18. Resultant Gain vs. Data Point - Actual Singlet Data



with the approximate singlet data. This is because the desired gain contours were originally generated using actual singlet data.

3.5.2.3 Computation Time Required

A major advantage to this method is that large dimension systems can be solved with relatively short computation time. Specifically, for the test using realistic desired contours, the following computation times were observed: for $M = 900$: $A^*BA = 1.5$ minutes CPU; $\underline{W}$ vector = 2 seconds CPU.

3.5.3 Test with Jammer Nulling

Of particular interest is the capability to null a specific area that may contain a jamming signal. To test this capability, it was decided to modify the desired contour pattern to include a null. The 11-dB closed contour in the vicinity of 1 degree azimuth/5 degrees elevation was arbitrarily chosen as the new null (see Figures 3-10 and 3-4). The minimum separation between this contour and the adjacent 14 dB contour is approximately 1 degree (or 44% of one beamwidth). Since the null area is of particular importance, the diagonal weighting matrix B of Equation 3.6 was employed (other than the identity matrix). The values of the diagonal elements of B were computed as an inverse function of the number of points defining the null contour.

3.5.3.1 Results for Required Nulling of 14 dB/degree

As an initial test of nulling with algorithm 1, the null contour level was set to 0 dB. Since the minimum separation between this contour and the adjacent 14 dB contour is approximately 1 degree, this represents a null "slope" of 14-dB/degree. The null contour is of particular importance and should be weighted more heavily in the determination of a beamweight vector used to match the desired contours. One method of weighting is to weight the null contour inversely proportional to the relative number of points defining it. Specifically, the null contour consists of 14 points out of a total of 900 defining the desired gain contours. In this case it was decided to weight the null contour equally as important as the combined importance of the remaining contours. That is, 50% of the total weighting was assigned to the null contour and 50% assigned to the remaining contours. Therefore, the value of the diagonal elements of the B matrix corresponding to the null contour points is given by 0.5/14. The value of the remaining diagonal elements of the B matrix corresponding to the remaining contour points is given by 0.5/886. The

Figure 3-19. Resultant 16 dB Contour - Actual Singlet Data

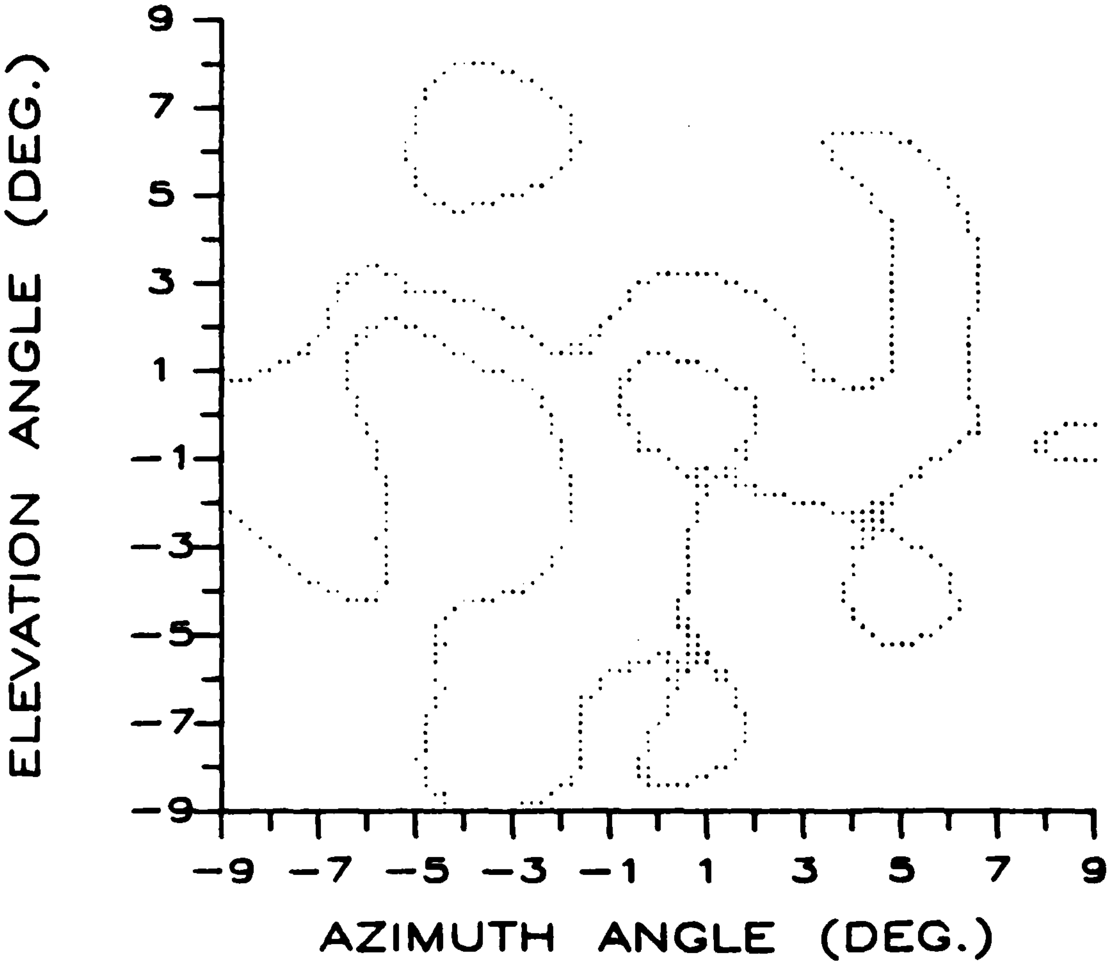


Figure 3-20. Resultant 14 dB Contour - Actual Singlet Data

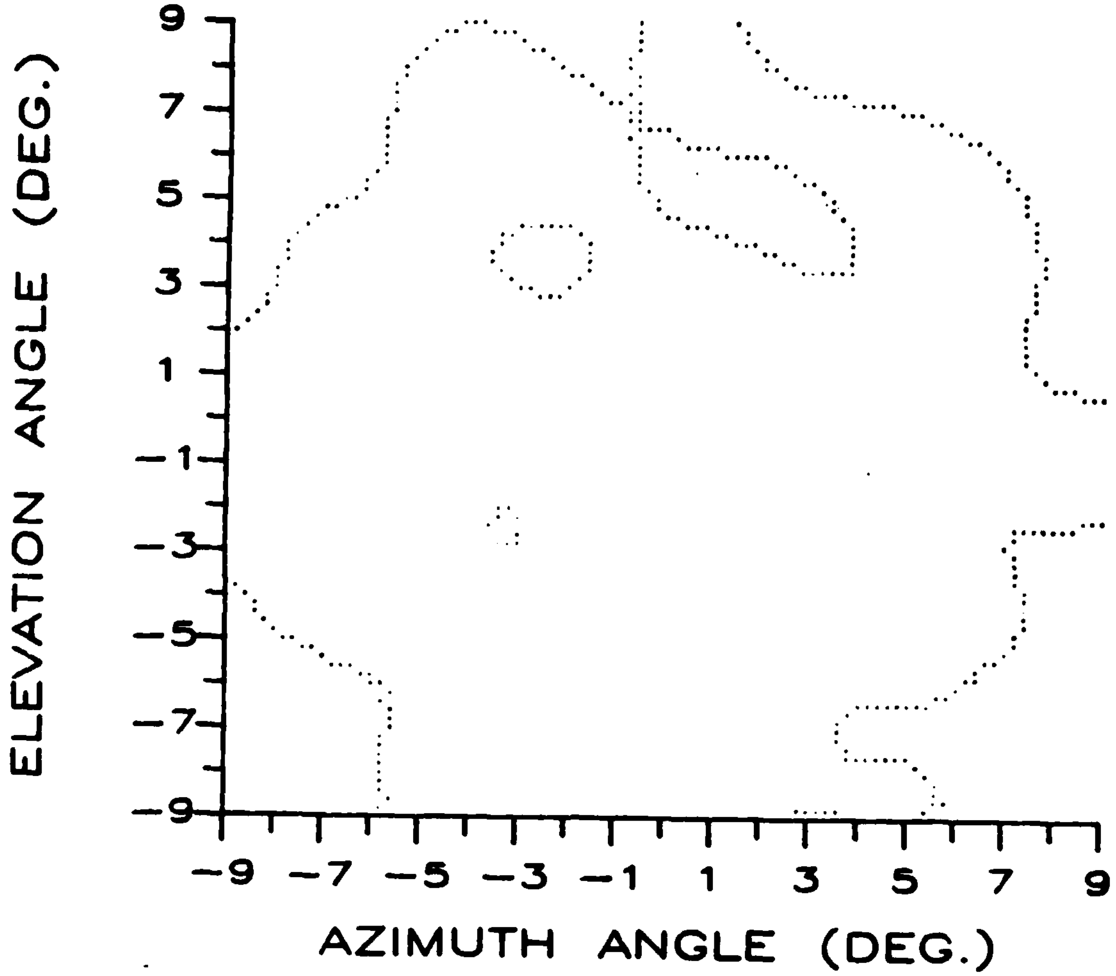
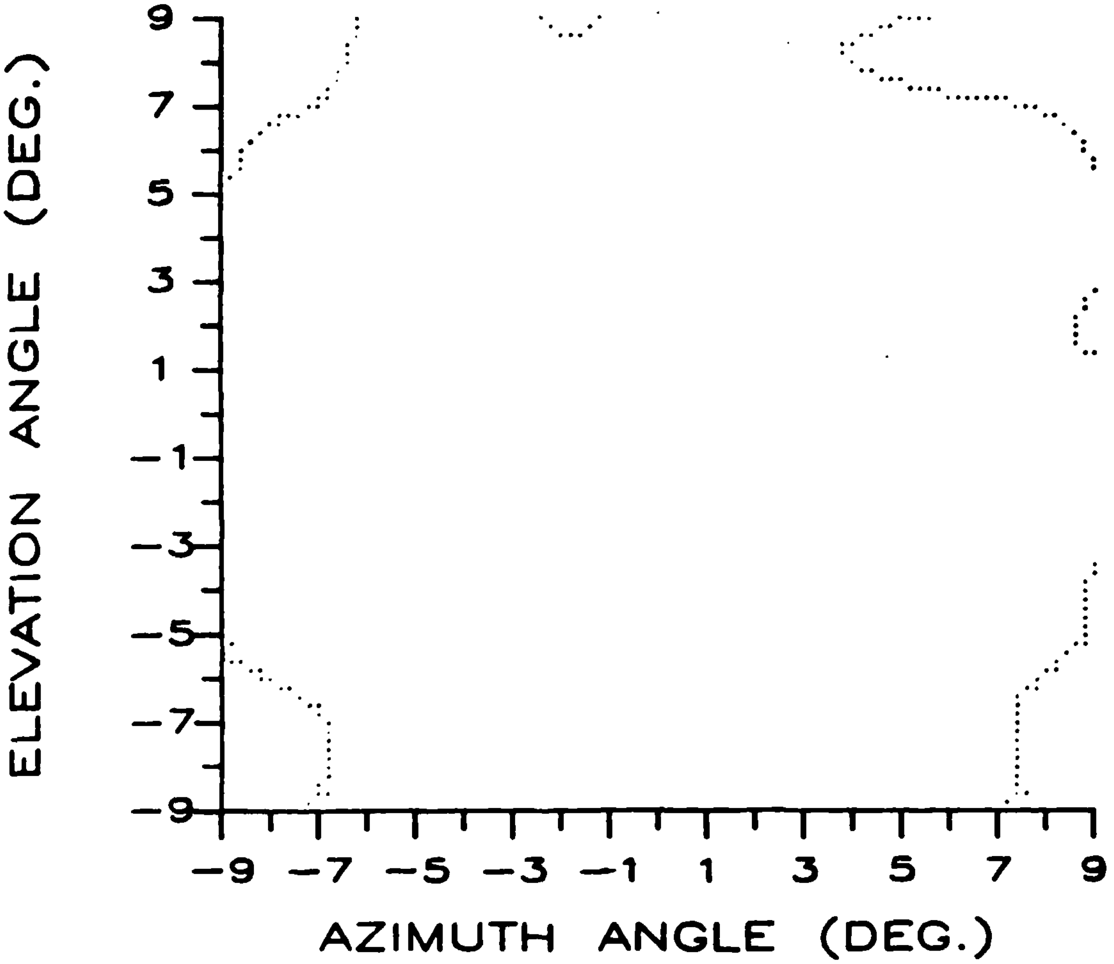


Figure 3-21. Resultant 11 dB Contour - Actual Singlet Data



beamweight vector was computed, and the resultant gain contours were determined as before. The beamweight vector is given in Table 3-2. The resultant 16-dB contour is shown in Figure 3-22. Very little change resulting from the effect of the jammer null was observed in this contour from that of Figure 3-19. The resultant 14 and 11 dB contours are shown in Figures 3-23 and 3-24, respectively. A noticeable distortion has occured in the vicinity of the jammer null on the 14-dB contour plot. This is due, in part, to the limited resolution capability of the MBA to discriminate between users and interferers separated by less than one beamwidth (here less than 44% of one beamwidth). In addition, a new 11-dB contour appears around the jammer null locations as expected. The desired and resultant 0 dB contours are shown in Figures 3-25 and 3-26, respectively.

3.5.3.2 Results for Required Nulling of 20 dB/deg.

To further test algorithm 1 with nulling, the null contour described above was reduced to -6 dB. This represents a null slope of approximately 20 dB/degree and a requirement of a ring null that may be near the theoretical maximum resolution capability of the MBR. The weighting on the null in terms of relative importance (B matrix) was increased by a factor of 10 over the 0 dB case described above.

The beamweight vector was computed and the resultant contours were drawn. The new beamweight vector for the -6 dB nulling case is given in Table 3-3. The resultant 16, 14 and 11-dB contours are shown in Figures 3-27 thru 3-29. A distortion similiar to that observed with the 0 dB nulling case above is also observed in these contours. The resultant 0 dB contour is shown in Figure 3-30 for reference with Figure 3-26. The desired and resultant -6 dB contours are shown in Figures 3-31 and 3-32, respectively. Only a single point was obtained for the desired -6 dB contour, as indicated by the arrow in Figure 3-32. This may be an indication of the resolution capabilities of the MBR to achieve the nulling depth specified in this case. Further work is recommended to determine if this result can be improved using phase tapering techniques.

Table 3-2. Computed Beamweight Vector for MBR Contour Pattern with 0 dB Null Contour - no scaling

BEAM NO.	REAL	IMAG.
1	0.0009822	0.0000155
2	0.0010547	-0.0000188
3	0.0011074	0.0000678
4	0.0008205	0.0000715
5	0.0009385	0.0000216
6	0.0007899	0.0001403
7	0.0011520	0.0001354
8	0.0010344	0.0000378
9	0.0011591	-0.0000974
10	0.0008333	0.0000611
11	0.0007858	0.0001163
12	0.0010016	0.0000774
13	0.0011327	-0.0000254
14	0.0010945	-0.0000003
15	0.0011035	-0.0000305
16	0.0008424	0.0000859
17	0.0012301	-0.0000819
18	0.0006837	0.0000556
19	0.0010566	0.0000108
20	0.0011484	-0.0001476
21	0.0009055	0.0001504
22	0.0011643	0.0000746
23	0.0012003	-0.0000264
24	0.0011289	0.0000416
25	0.0009066	0.0000059
26	0.0009029	-0.0000157
27	0.0013783	0.0000723
28	0.0011284	-0.0000040
29	0.0010261	0.0000809
30	0.0010361	-0.0000832
31	0.0009513	0.0000987
32	0.0009280	-0.0001767
33	0.0011828	-0.0000013
34	0.0010419	-0.0000054
35	0.0011403	-0.0000002
36	0.0009614	0.0001086
37	0.0011550	-0.0001084
38	0.0010194	-0.0000446
39	0.0011663	-0.0000246
40	0.0014861	0.0000548
41	0.0010782	0.0001515
42	0.0010355	-0.0000190
43	0.0007815	0.0002940
44	0.0007633	0.0000214
45	0.0010299	0.0000765
46	0.0009337	0.0000723
47	0.0004661	0.0002049
48	0.0005515	0.0002970

BEAM NO.	REAL	IMAG.
49	0.0011890	-0.0000339
50	0.0008801	0.0000030
51	0.0008132	0.0001186
52	0.0014282	-0.0000519
53	0.0010347	-0.0000529
54	0.0006743	0.0000147
55	0.0012186	0.0000335
56	0.0008794	0.0001491
57	0.0008233	0.0000738
58	0.0008948	0.0003108
59	0.0010461	-0.0002131
60	0.0006471	-0.0000587
61	0.0007627	0.0006237

Figure 3-22. Resultant 16 dB Contour

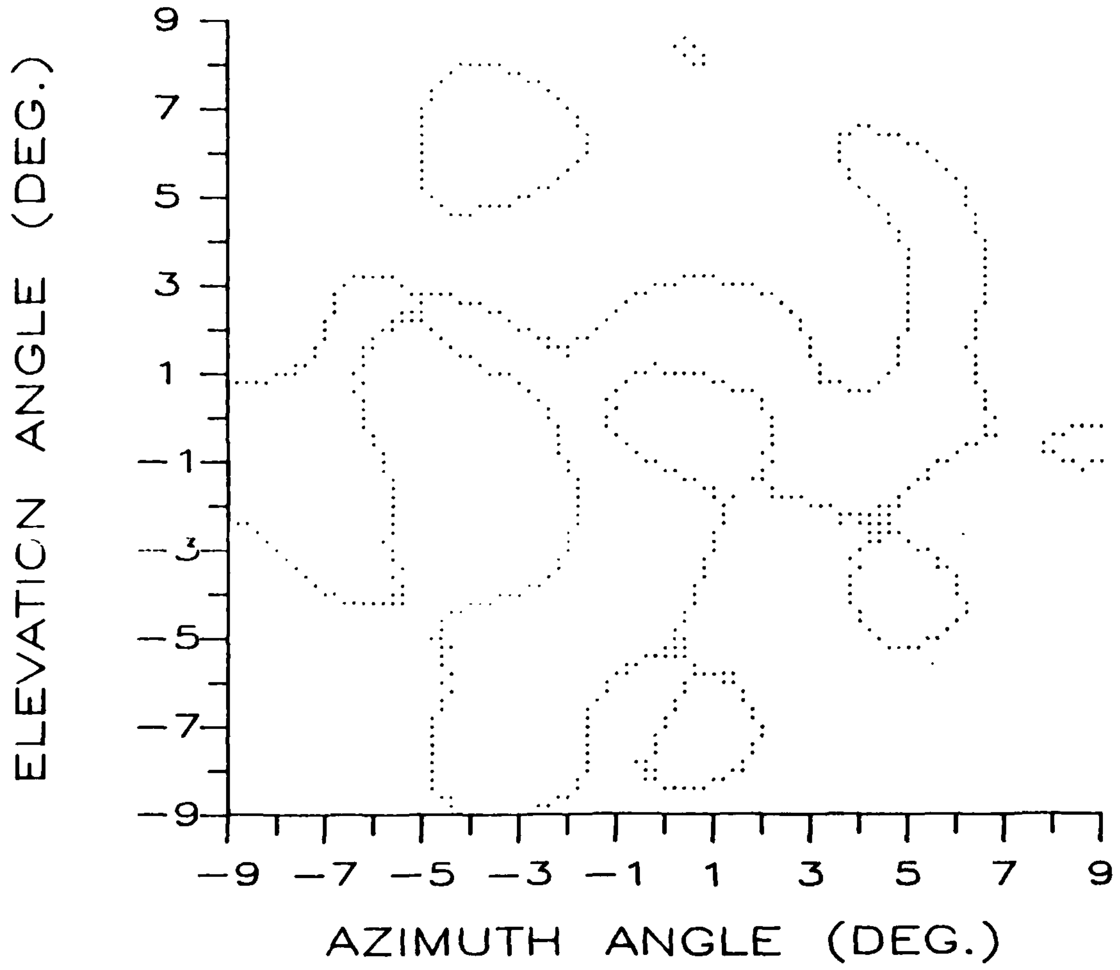


Figure 3-23. Resultant 14 dB Contour

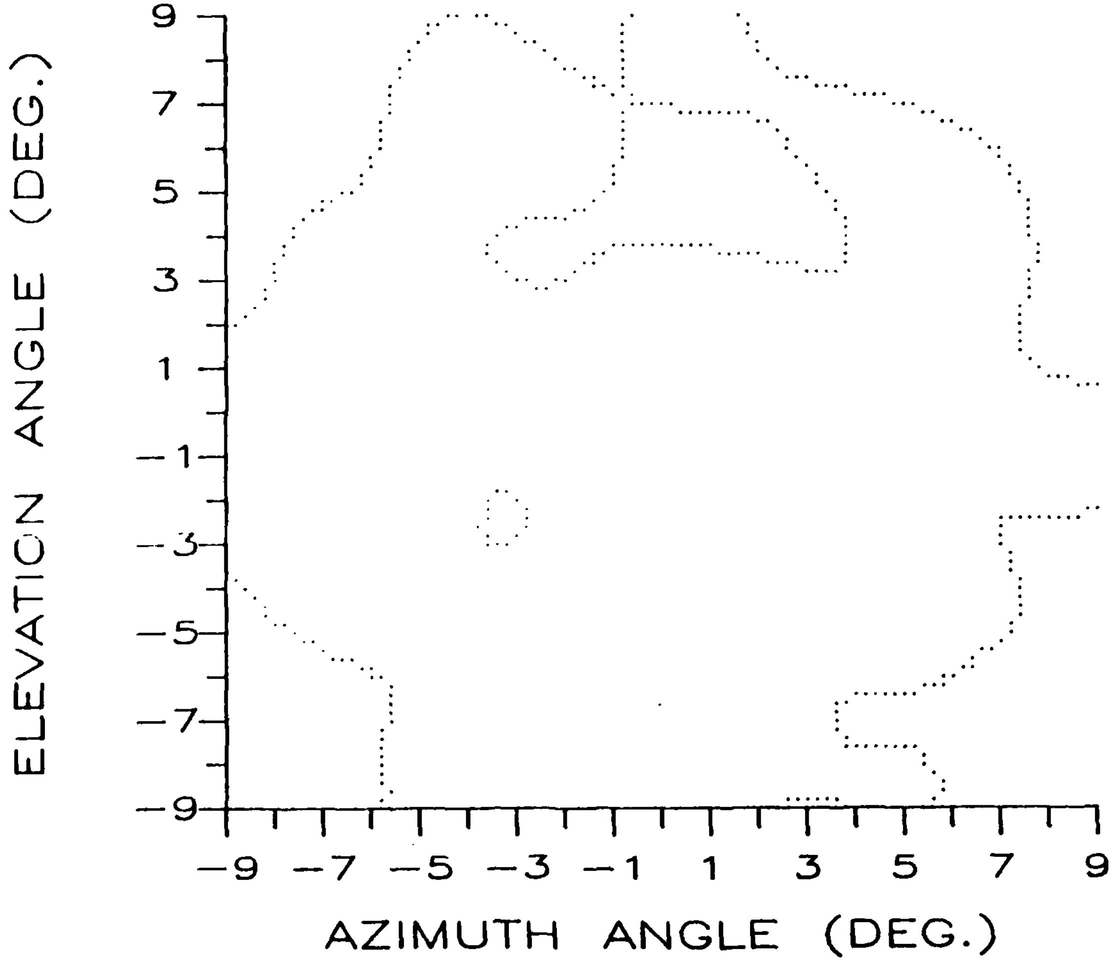


Figure 3-24. Resultant 11 dB Contour

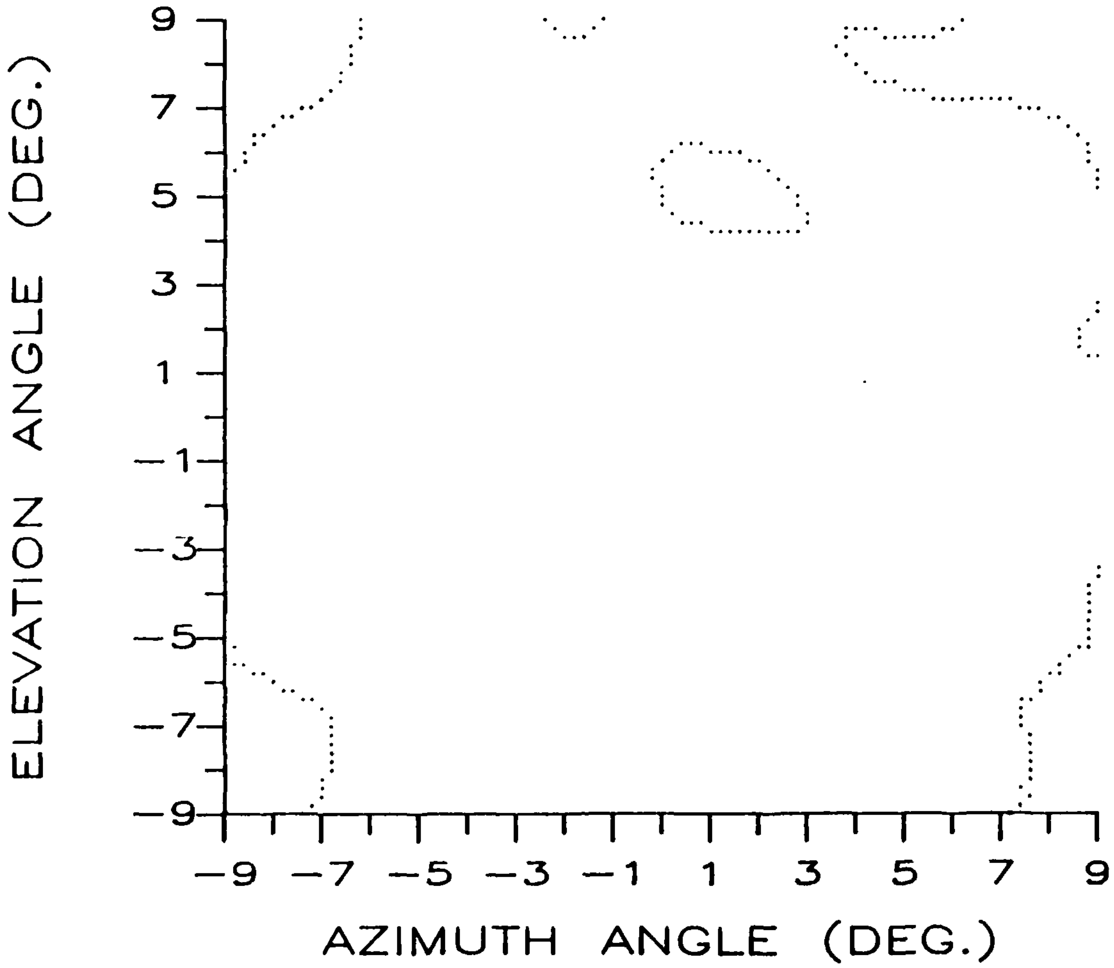


Figure 3-25. Desired 0 dB Contour

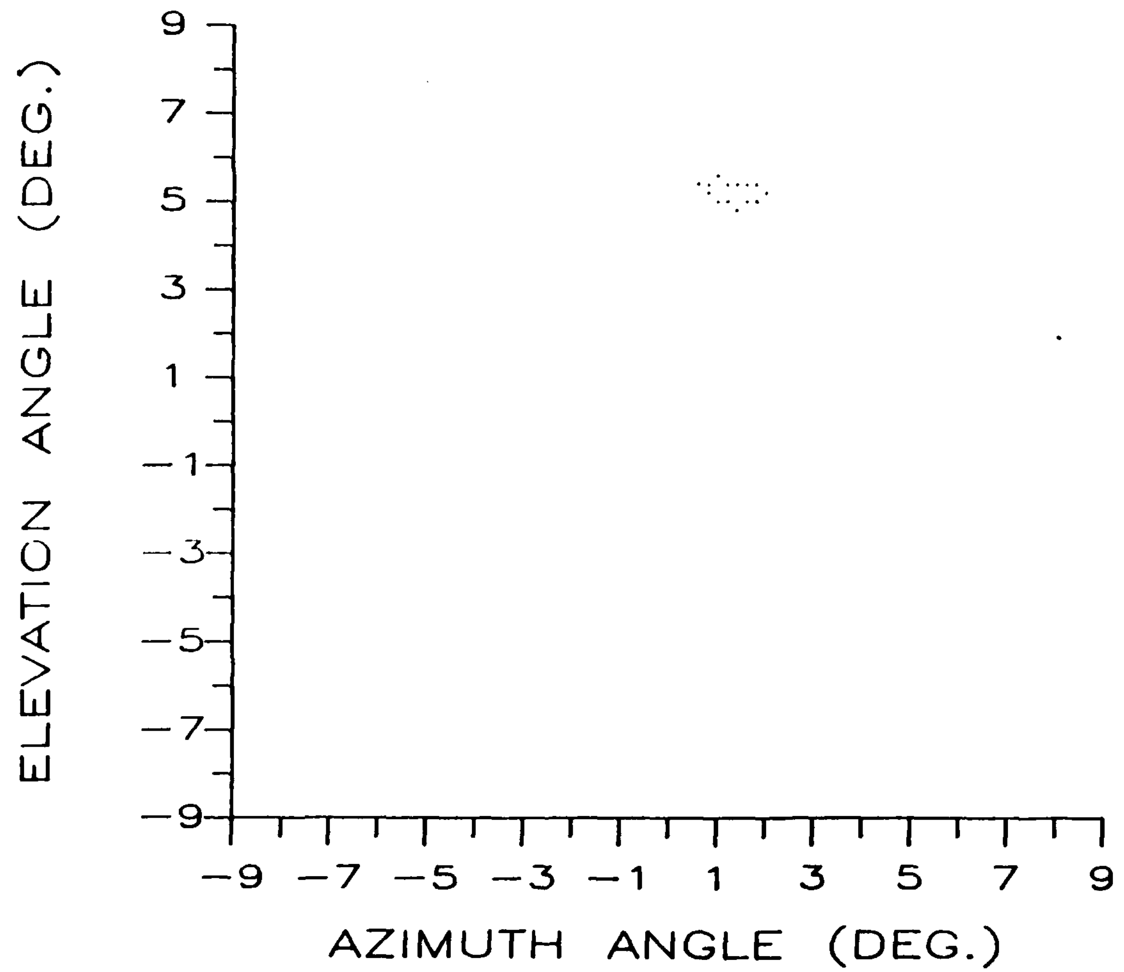


Figure 3-26 Resultant 0 dB Contour

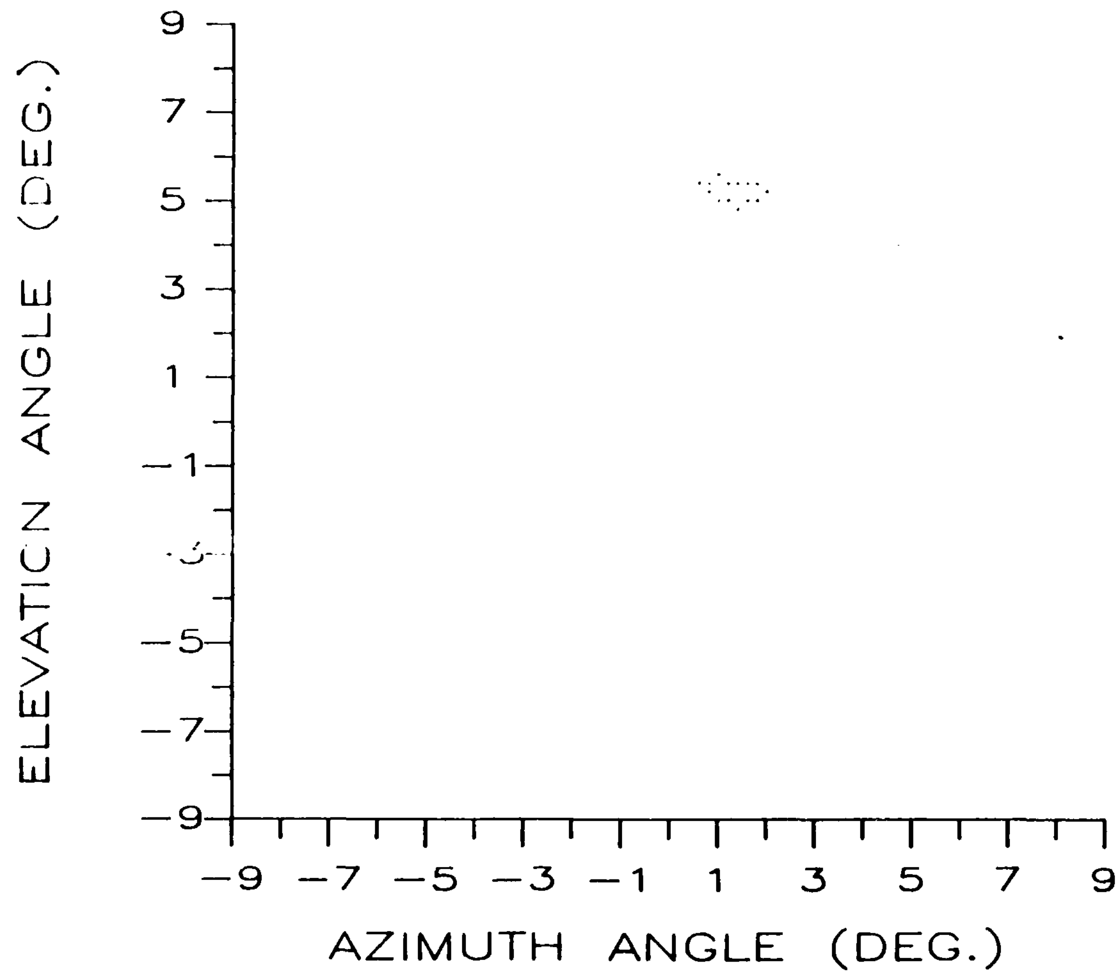


Table 3-3. Computed Beamweight Vector for MBR Contour Pattern with -6 dB Null Contour - no scaling

BEAM NO.	REAL	IMAG.	BEAM NO.	REAL	IMAG.
1	0.0009866	0.0000029	48	0.0003847	0.0002422
2	0.0010864	-0.0000256	49	0.0011978	-0.0000990
3	0.0011173	0.0000822	50	0.0008470	0.0000374
4	0.0007935	0.0001345	51	0.0008031	0.0000839
5	0.0009666	0.0000492	52	0.0014658	-0.0001177
6	0.0007774	0.0001120	53	0.0009403	0.0000687
7	0.0011327	0.0001098	54	0.0004690	0.0002359
8	0.0009970	0.0000510	55	0.0010895	0.0001341
9	0.0012004	-0.0001517	56	0.0009483	0.0001695
10	0.0008207	0.0000529	57	0.0008236	0.0001543
11	0.0007496	0.0001111	58	0.0009101	0.0002669
12	0.0009792	0.0000767	59	0.0011247	-0.0004103
13	0.0011512	0.0000169	60	0.0004667	-0.0001720
14	0.0011286	-0.0000226	61	0.0006813	0.0005476
15	0.0011049	-0.0000396			
16	0.0007291	0.0001486			
17	0.0012621	-0.0000762			
18	0.0006555	0.0000717			
19	0.0010660	0.0000253			
20	0.0011505	-0.0001747			
21	0.0008870	0.0001331			
22	0.0011201	0.0001330			
23	0.0012796	-0.0000295			
24	0.0011410	0.0000391			
25	0.0009030	0.0000267			
26	0.0009133	-0.0000441			
27	0.0014045	0.0000132			
28	0.0010822	-0.0000008			
29	0.0010451	0.0001260			
30	0.0010560	-0.0001369			
31	0.0008559	0.0000883			
32	0.0008707	-0.0002431			
33	0.0011944	-0.0000209			
34	0.0010399	-0.0000265			
35	0.0010323	-0.0000179			
36	0.0010064	0.0000965			
37	0.0011683	-0.0001438			
38	0.0010315	-0.0000697			
39	0.0012175	0.0000427			
40	0.0016190	0.0001375			
41	0.0010620	0.0002229			
42	0.0010273	0.0000427			
43	0.0007603	0.0002423			
44	0.0007292	0.0000379			
45	0.0010208	0.0001435			
46	0.0009285	0.0000261			
47	0.0001733	0.0001368			

Figure 3-27. Resultant 16 dB Contour - -6 dB null at 1 deg. azimuth / 5 deg. elevation

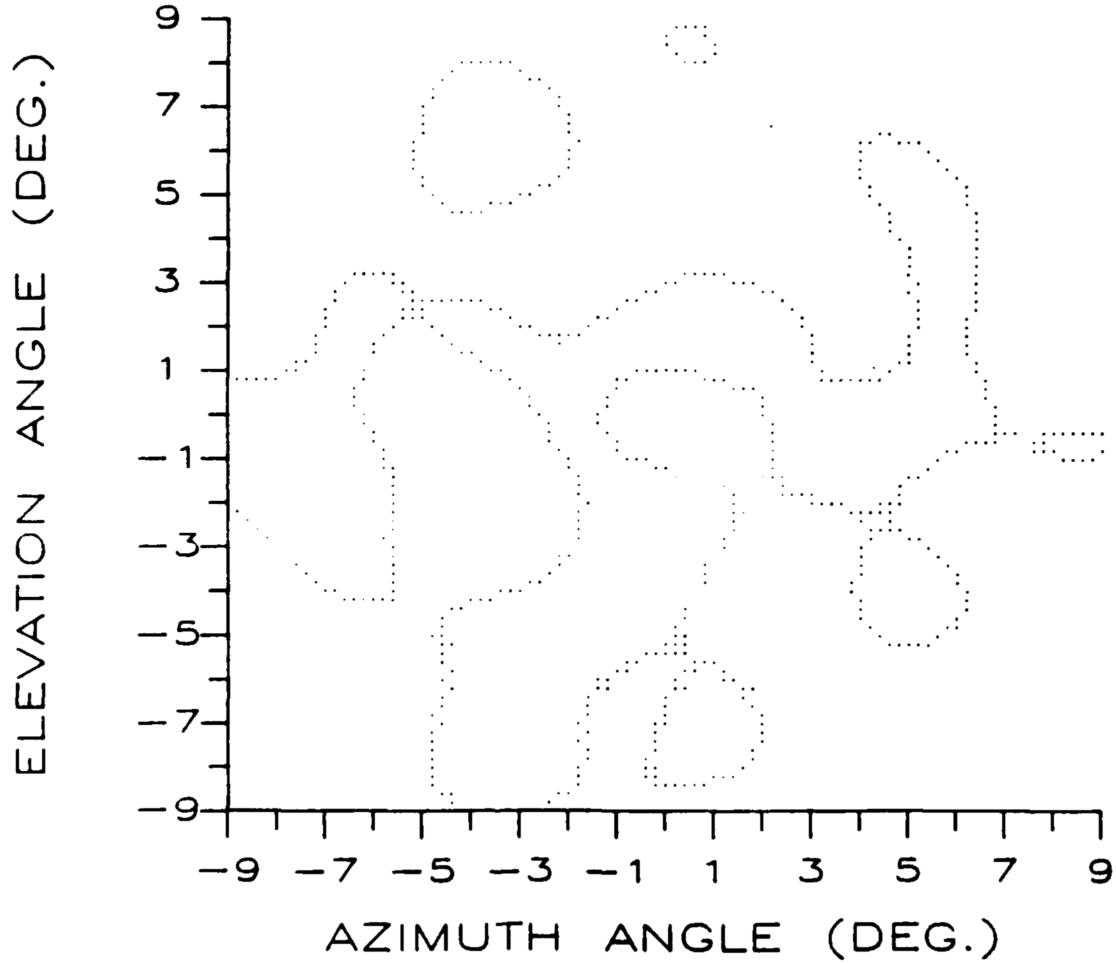


Figure 3-28. Resultant 14 dB Contour - -6 dB null at 1 deg. azimuth / 5 deg. elevation

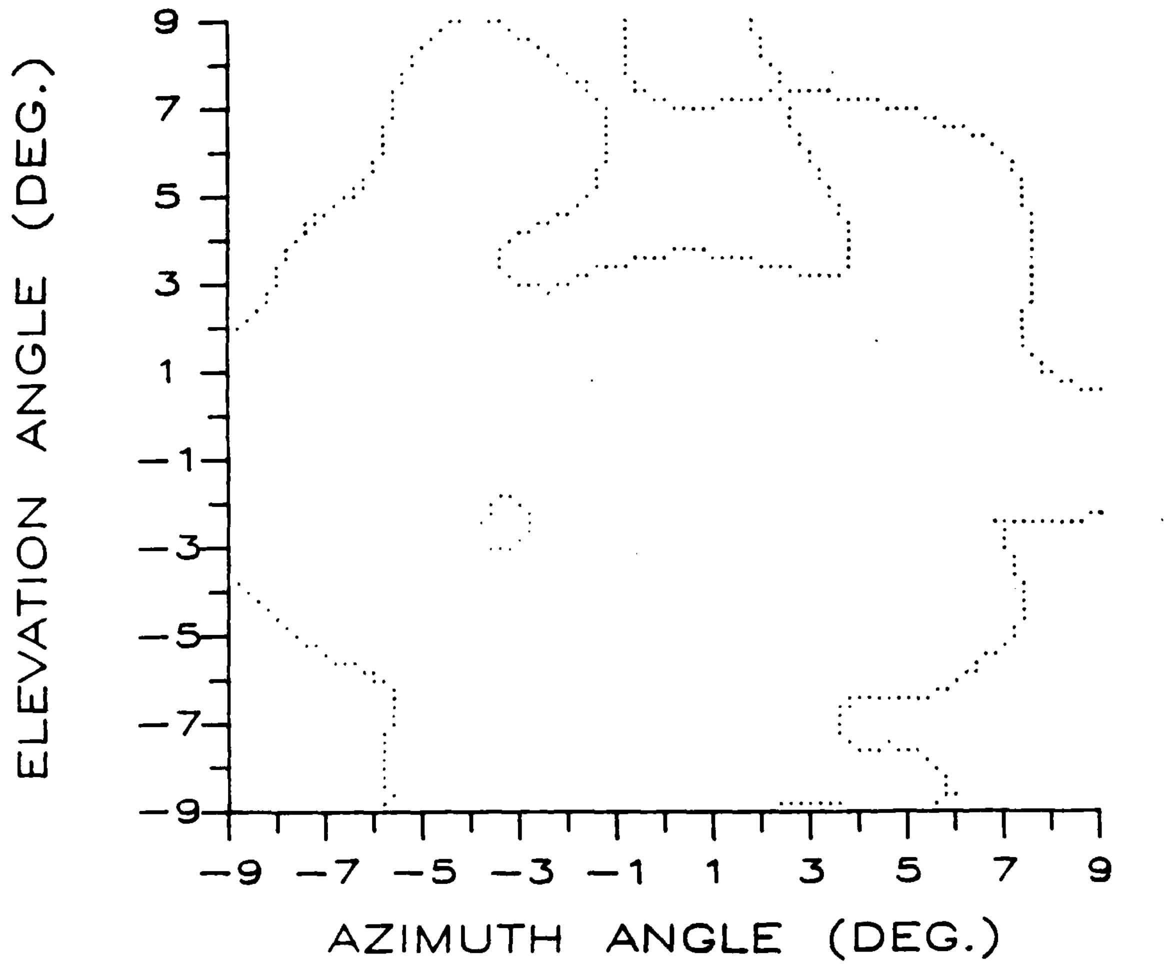


Figure 3-29. Resultant 11 dB Contour - -6 dB null at 1 deg. azimuth / 5 deg. elevation

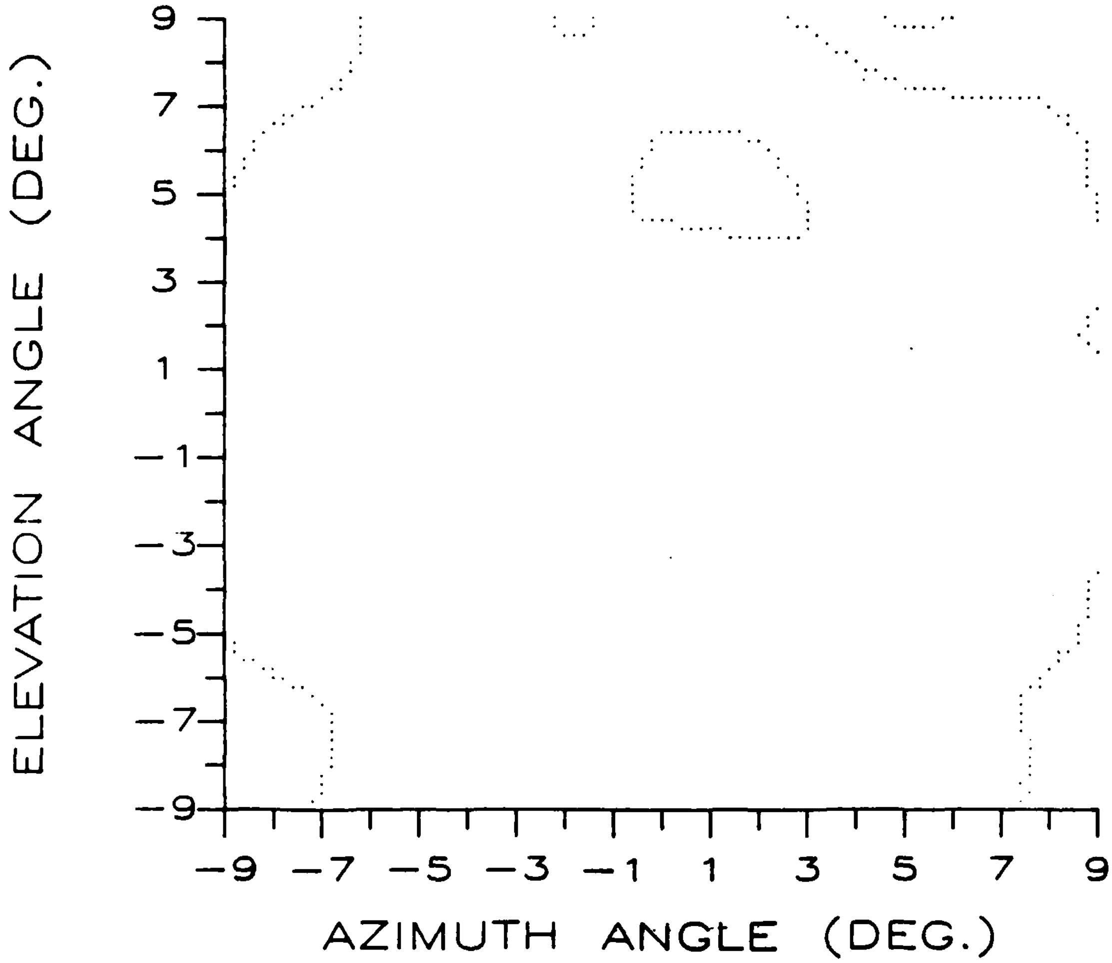


Figure 3-30. Resultant 0 dB Contour - -6 dB null at 1 deg. azimuth / 5 deg. elevation

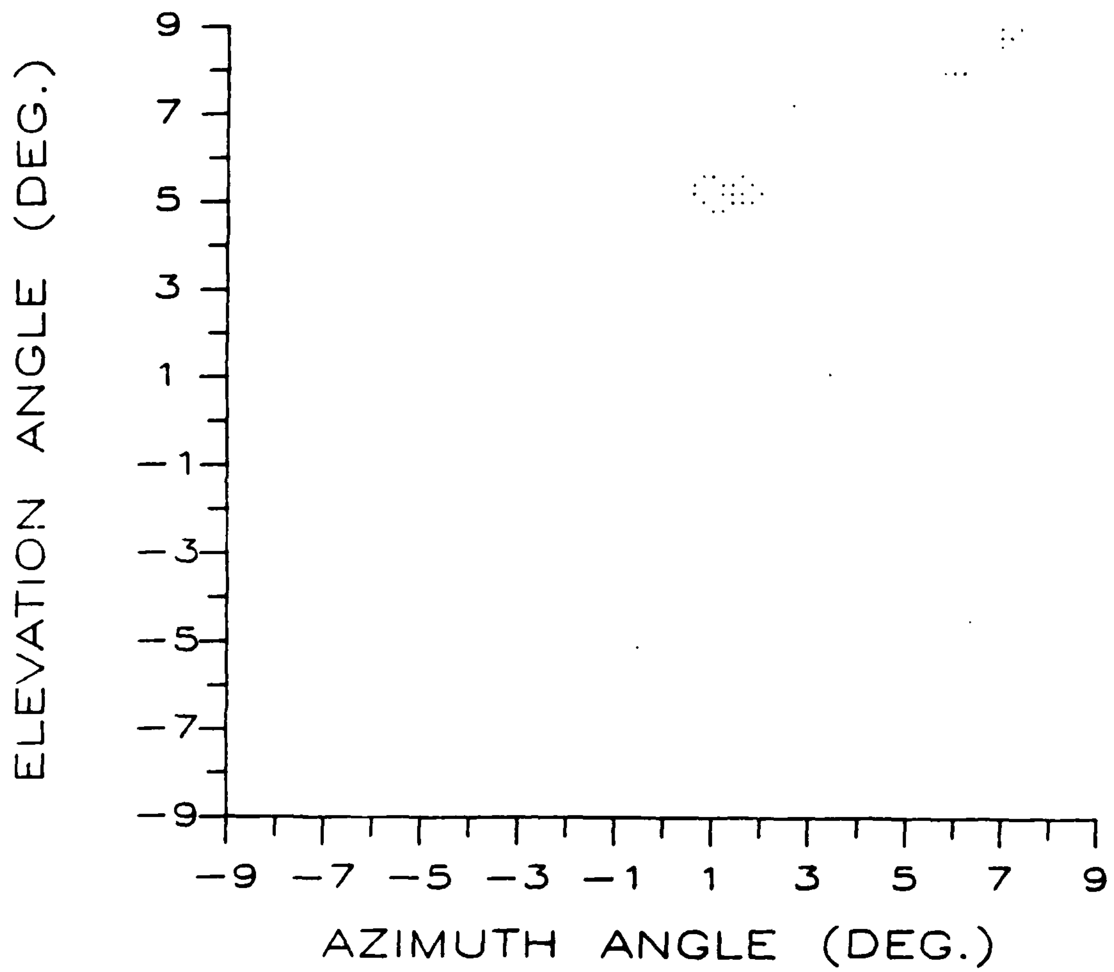


Figure 3-31   Desired − 6 dB Contour

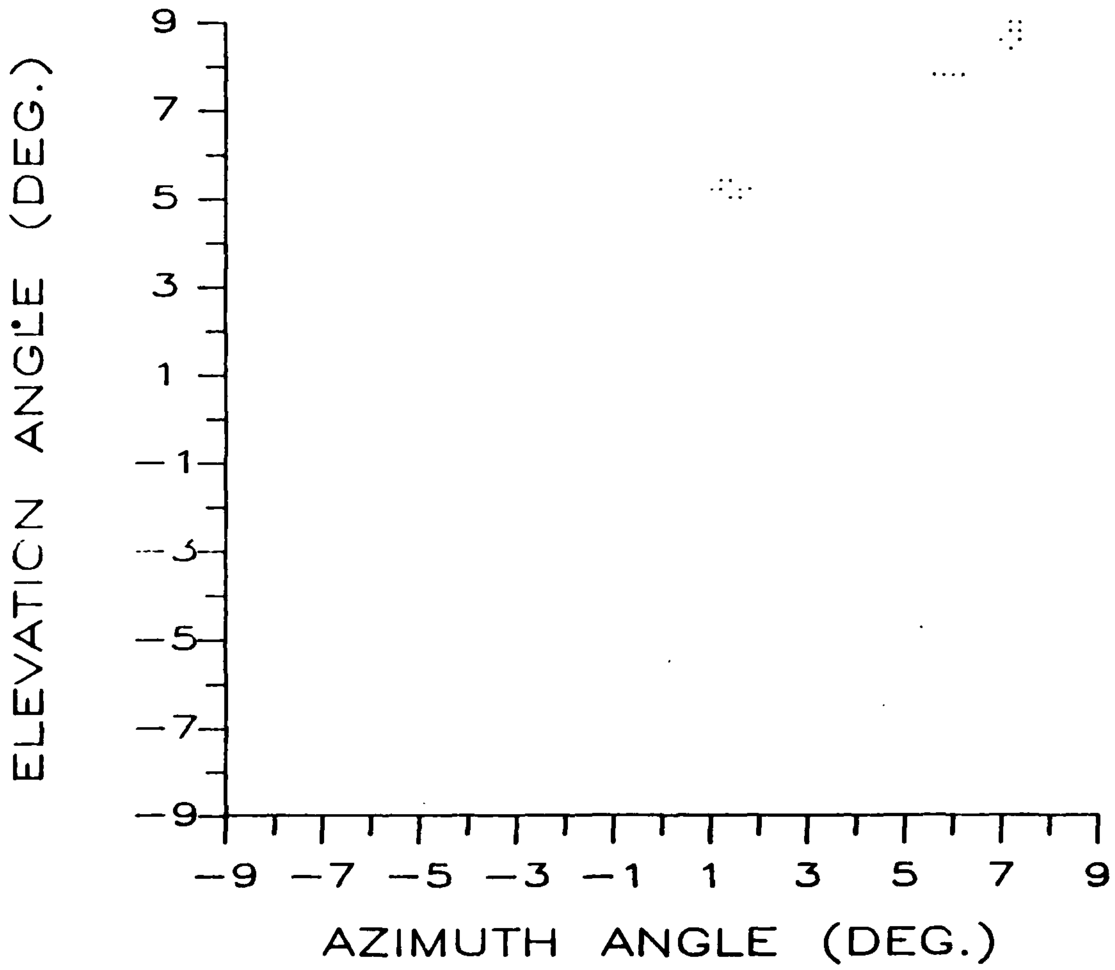
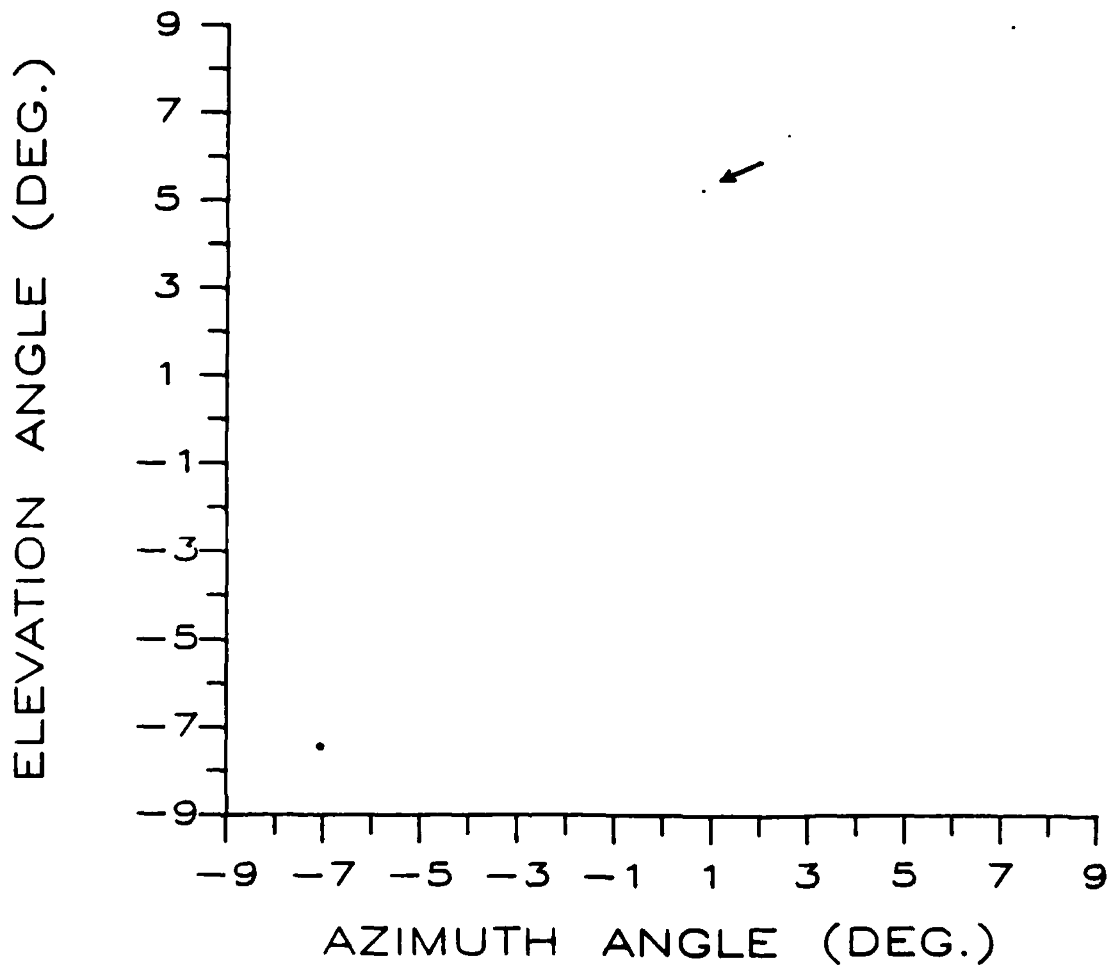


Figure 3-32. Resultant -6 dB Contour



3.6 VERIFICATION OF ALGORITHM 2

The MSE solution for magnitude-only gain contours has been implemented in software. Initial tests have been performed using the MBR contour pattern described in Section 3.5.2 as the desired contour and the solution beamweight vector determined by algorithm 1 as the initialization for the Davidon-Fletcher-Powell algorithm.

3.6.1 Results using Beamweight Solution from Algorithm 1 - Non-Nulling Scenario

The beamweight vector solution obtained using algorithm 1 for the non-nulling scenario with actual singlet data (Table 3-1) was used as input to the Davidon-Fletcher-Powell algorithm. By providing an initial guess to the algorithm that is close to the desired solution, computational time can be reduced and an improved solution is obtained. Due to the increased computational time involved in applying this algorithm, it is more suitable for "fine tuning" of resultant contour patterns to meet some desired requirements. A beamweight vector for the desired MBR contour pattern was obtained and is given in Table 3-4. The resultant 16, 14 and 11 dB contour patterns are shown in Figures 3-33 thru 3-35. Significant improvement can be seen in the resultant contours over the corresponding contours of Figures 3-19 thru 3-21.

3.6.2 Computation Time Required

Since this method involves a gradient search technique, the computational time is significantly longer than with algorithm 1. Specifically, for the scenario described above the CPU time required to arrive at a solution was 15 minutes 55.8 seconds using the solution beamweight vector determined by algorithm 1 as the initialization (seed value) for the Davidon-Fletcher-Powell algorithm.

3.7 CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

3.7.1 Conclusions

The approach pursued here for determining the beamweight vector $\underline{W}$ (i.e., *algorithm 1) looks promising for use in the DNPS because of the speed with which a* solution is obtained for a given singlet data matrix. As stated previously, the required weight vector $\underline{W}$ computation of A^*BA needs to be performed only once even if the contour gain levels are changed, as long as contour area definitions and weighting matrix

Table 3-4. Computed Beamweight Vector for MBR Contour Pattern after Functional Minimization – no scaling

BEAM NO.	REAL	IMAG.	BEAM NO.	REAL	IMAG.
1	0.0009514	-0.0000077	49	0.0012163	-0.0000611
2	0.0010177	-0.0000172	50	0.0008973	0.0000408
3	0.0011505	0.0000312	51	0.0007441	0.0001714
4	0.0008298	0.0000301	52	0.0015574	-0.0000304
5	0.0009089	-0.0000190	53	0.0009979	0.0000585
6	0.0007561	0.0001220	54	0.0011487	-0.0001652
7	0.0012142	0.0001448	55	0.0012163	-0.0001508
8	0.0010451	0.0000569	56	0.0008904	0.0001561
9	0.0011122	-0.0000715	57	0.0008420	0.0000556
10	0.0008494	0.0000864	58	0.0008748	0.0004081
11	0.0008416	0.0001863	59	0.0010373	-0.0002389
12	0.0010035	0.0000592	60	0.0007126	0.0001688
13	0.0011497	0.0000146	61	0.0007730	0.0005137
14	0.0010512	-0.0000261			
15	0.0011395	-0.0000195			
16	0.0009009	0.0001224			
17	0.0012454	-0.0001423			
18	0.0006870	-0.0000036			
19	0.0010462	-0.0000276			
20	0.0011622	-0.0000802			
21	0.0009509	0.0001797			
22	0.0011630	-0.0000132			
23	0.0010796	-0.0000373			
24	0.0011037	0.0000332			
25	0.0009284	-0.0000026			
26	0.0008916	-0.0000251			
27	0.0012490	0.0000560			
28	0.0011868	-0.0000501			
29	0.0010265	0.0001511			
30	0.0010295	-0.0000817			
31	0.0011089	0.0000985			
32	0.0010424	-0.0000662			
33	0.0011877	0.0000343			
34	0.0010449	-0.0000142			
35	0.0011651	0.0000506			
36	0.0009360	0.0000824			
37	0.0011461	-0.0000997			
38	0.0011270	-0.0000910			
39	0.0010940	0.0000282			
40	0.0012355	-0.0000631			
41	0.0010116	0.0001015			
42	0.0010793	-0.0000670			
43	0.0007474	0.0003713			
44	0.0006865	-0.0000558			
45	0.0011065	0.0000830			
46	0.0009118	0.0002233			
47	0.0009653	0.0001246			
48	0.0009527	0.0003428			

Figure 3-33. Resultant 16 dB Contour - Algorithm 2

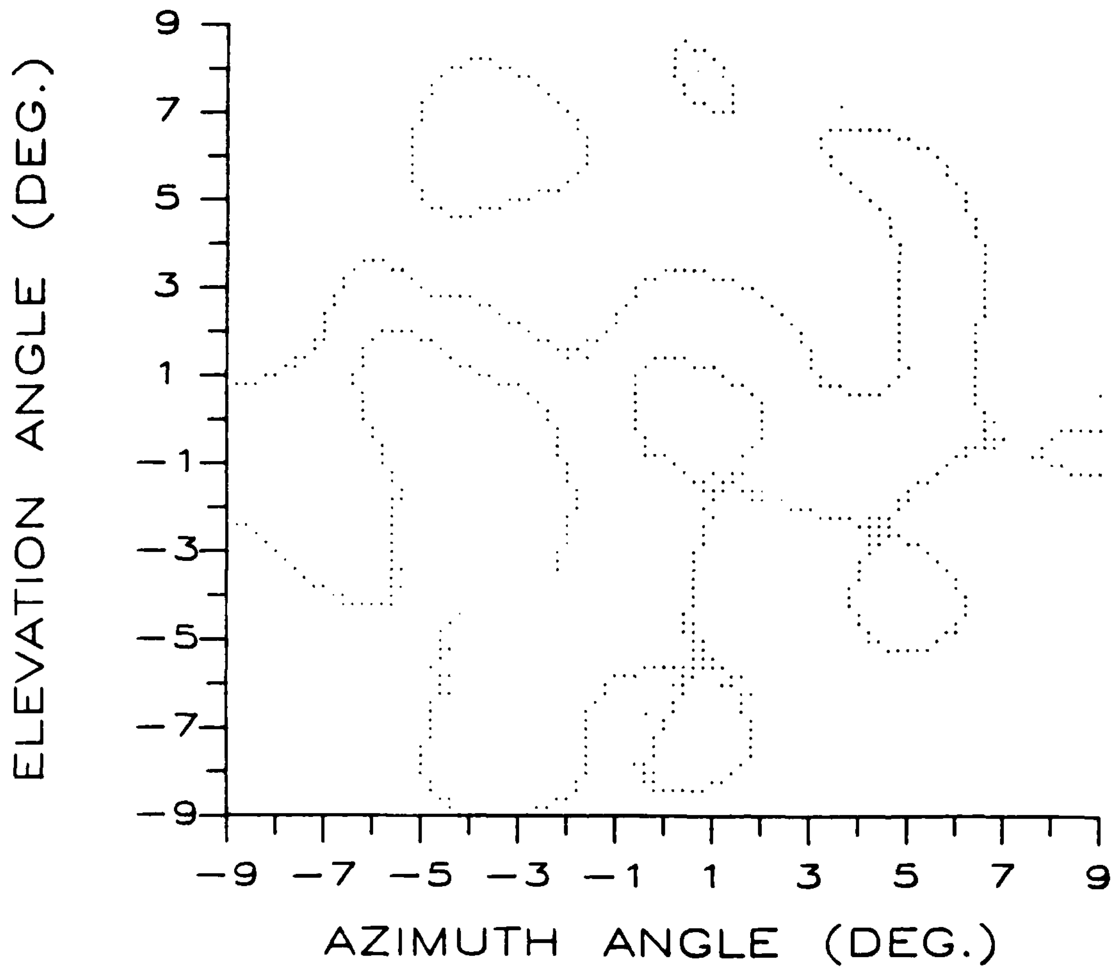


Figure 3-34. Resultant 14 dB Contour - Algorithm 2

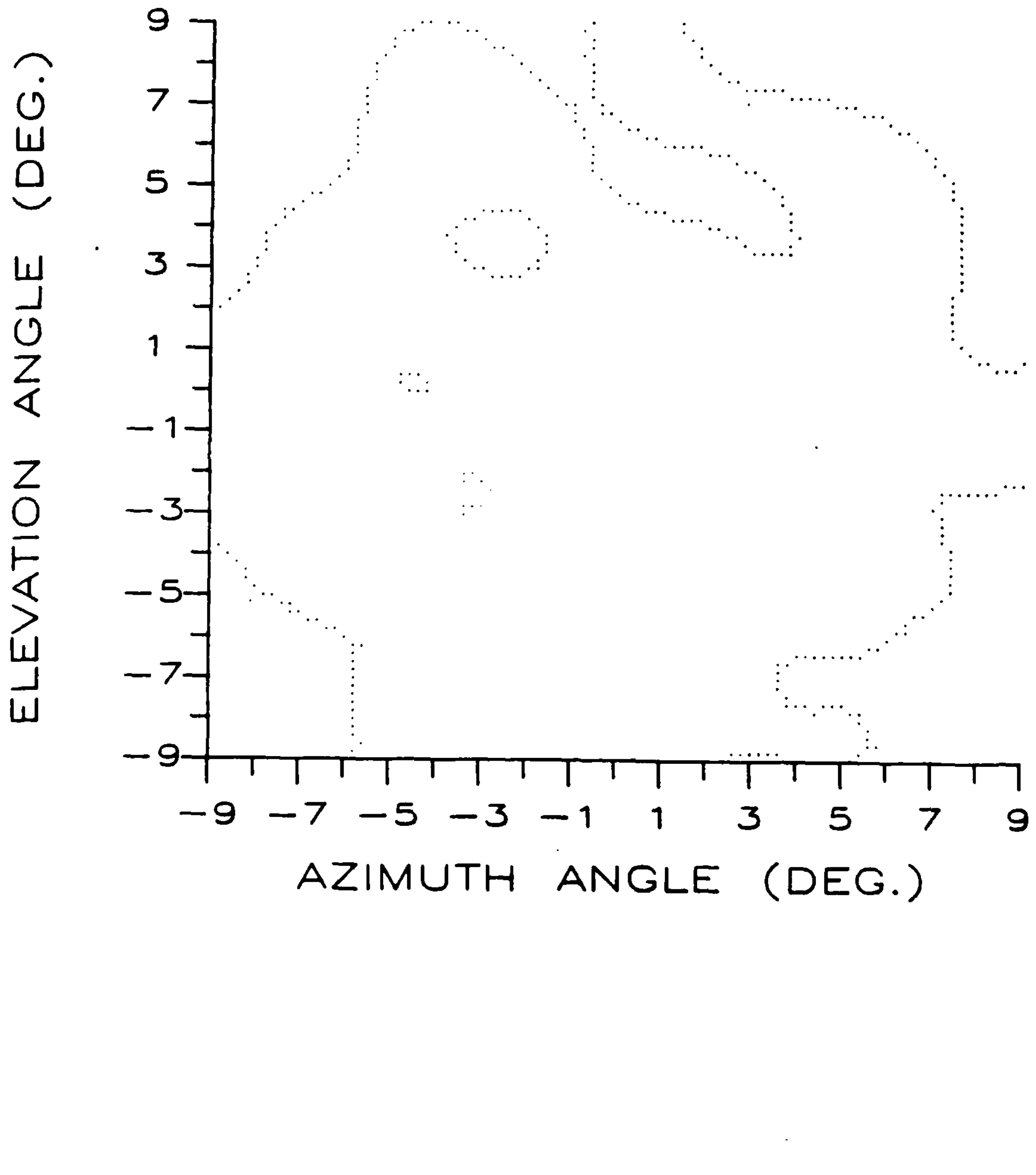
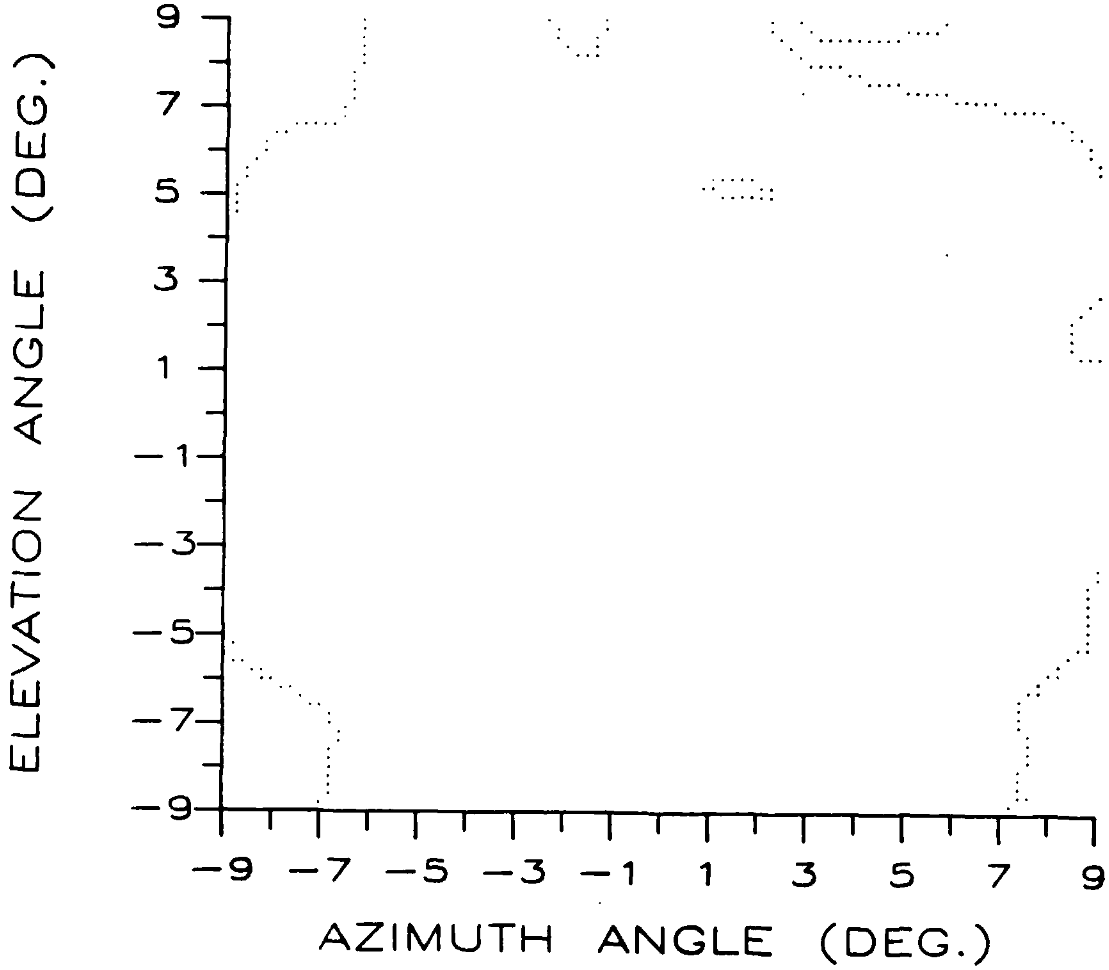


Figure 3-35. Resultant 11 dB Contour - Algorithm 2



B remain fixed for a given set of geographic contours. Large systems can be solved with very good results, as shown. If contour area definitions are changed in only a small area, a partial beamweight solution could be performed for the area in question. The resulting beamweights could be inserted into the existing $\underline{W}$ vector to obtain the new resultant contours.

3.7.2 Suggestions for Future Work

It is recommended that further testing of algorithm 1 using measured singlet data be performed. The issue of generating a partial beamweight solution for small contour areas that have undergone a change in requirements could be investigated. Also, algorithm 1 will be of further interest for tests with phase tapering, since the solution considers both magnitude and phase of the gain.

The functional minimization approach (algorithm 2) can further be tested with the nulling scenarios described above to improve upon the results obtained thus far.

Appendix A

Test Plan/Report

PREVIOUS PAGE IS BLANK

A.1 Overview

The DOCS has been enhanced by the addition of the adaption subsystem to the DNPS. The adaption subsystem produces the largest supportable subscenario of the active DNPS scenario. The subsystem generates network configurations that satisfy arbitrary jammer, terminal excessive power requirements and/or equipment failure constraints, in accordance with the user-specified adaption criteria. The active scenario (the input to the adaption subsystem) contains definitions of terminals, frequency hop links, FDMA links, ECCM links/circuits, jammers, subnets, satellite characteristics and the adaption criteria. The subscenario (output of the adaption subsystem) is based on the combination of constrained traffic, prioritized ranking of unconstrained traffic elements and unconstrained traffic processing (an iterative process).

To support this integration the following areas required changes to, or added interfaces with, DOSS/DNPS:

- On-Line Static Database Loader
- Scenario definition
- Traffic adaption
- Report generation.

Two major requirements were added to the existing DOSS by the ENAM:

- The capability to temporarily ignore traffic elements in the network calculations
- The capability to constrain traffic elements into the network, regardless of power requirements.

A major goal in the addition was to provide the capability to determine the maximum supportable network, given a set of operator-entered constraints. The Traffic Adaption subsystem was added to support this requirement.

A.2 Purpose of Testing

The major goal in the testing effort is the verification of output continuity in the transition from DOSS version 2.1 to DNPS version 3.1. This involved identification of a typical user scenario and generation of output from the subsystems of both versions of the software. The output is analyzed to determine if the new software is still producing acceptable results. Should the software produce inconsistent output, an effort will be made to identify the problem and, if possible, create a 'work around' solution (one that does not require the software to be modified). An additional goal in the testing effort is the establishment of baseline testing procedures and scenarios for the ECCM Adaption subsystem. These testing procedures for output data continuity and the ECCM Adaption subsystem will aid in the future transitions between version of operational software.

A.3 Testing Sequence Overview

The following steps are typical of the testing process. Specific scenarios will modify the required test steps because of the data contained in the scenario. An example of this is 'jammer nulling'; the menu that invokes this algorithm will be displayed only when a jammer is present in the scenario being tested. The specific test steps required for each scenario are detailed below.

1. **Retrieve Scenario** - Loads the desired scenario to test into the active database.

2. **Analysis Support** - Calculates the directive gains for the earth coverage antennas, earth-relative and satellite-relative terminal/jammer azimuth and elevation, slant ranges to terminals, path loss (based on the center frequency), argument of perigee, semi-major axis and range rate data (when the sub-point is calculated).

    - Initialize constants

    - Verify satellite location

    - Start analysis

3. **Allocation** - Generates or selects all the required network parameters for the network performance analysis.

    - All options off except summary report and GDA pointing

- Start allocation

4. **Network Performance** - Performs the forward link calculations to support the defined scenario.

    - All options off except summary report

    - Start network performance processing.

5. **Report Generation** - Generates various types of reports of the expected performance of the active scenario. The following reports are typical:

    - All links and all channels

    - Satellite analysis

    - Terminal directive gains

    - Beam weights

6. **Adaption** - Generates network configurations that satisfy arbitrary jammer, excessive power requirements and/or equipment failure constraints, in accordance with the user-specified adaption criteria Adaption is run when the active scenario cannot achieve a realizable network.

    - Use allocation beam weights

    - All default settings

    - Select/Modify Operator Interrupt Criteria

        a. Start Adaption
        Create the following reports :

        o *Traffic Ranking Report*

        o Network Performance

        o ECCM Link Summary

        o Satellite Analysis

        Verify 'CCC' are included in the network.

        b. Continue with Adaption
        Create the following reports :

- o Traffic Ranking Report
- o Network Performance
- o ECCM Link Summary
- o Satellite Analysis

Verify 'IET' are included in the network.

c. Continue with Adaption
Create the following reports :

- o Traffic Ranking Report
- o Network Performance
- o ECCM Link Summary
- o Satellite Analysis

Verify 'RLL' are included in the network.

- Set breakpoint option 1

a. Continue with Adaption
Create the following reports :

- o Traffic Ranking Report
- o Network Performance
- o ECCM Link Summary
- o Satellite Analysis

A.4 Test Scenario One

A.4.1 Scenario Description

Test scenario one included ECCM links, FH traffic, FDMA links and a definition for a jammer. A single user group was defined and the priority assigned to this group was '000002.' The ECCM circuits of the ECCM links were established with priorities of '000002.' Data rates and circuits within the ECCM links were varied to exercise the software. FDMA links were spread between channels 1, 2, 3 and 4. Modulation and coding type also were varied. The number of FH links was small and were included for completeness. The adaption criteria established were of the same priority (regardless of criteria type) and of the same name.

A.4.2 Scenario One Test Steps

Input	Description	Output
Username	Enter username	Password prompt
Password	Enter password	DOSS/DNPS Top level menu
4	Scenario Definition	Scenario Definition Menu
9	Mutiple Scenarios	Catalog Of Saved Scenarios
2	Retrieve A Scenario	Prompt To Select A Scenario
5	Index Of Scenario	Options For Scenario Loading
8	Retrieve Entire Scenario	Database Reset Prompt
Y	Verifies Database Reset	Scenario Catalog Menu
Level-Up	Return To Scenario Definition Menu	Scenario Definition Menu
Level-Up	Return To Top Level DNPS Menu	Top Level DNPS Menu
5	Analysis Support	Analysis Support Menu
1	Initialize Satellite Constants	Satellite IRON Prompt
6451	The IRON Of The Satellite	Analysis Support Menu
4	List/Modify Satellite SSP	Satellite Subpoint Position Detail
4	Satellite Longitude	Logitude Prompt
-135	Enter Longitude For Satellite	Satellite Subpoint Position Detail
Level-Up	Return To Analysis Support Menu	Analysis Support Menu
5	Perform Satellite Analysis	Analysis Support Menu
Level-Up	Return To Top Level DNPS Menu	Top Level DNPS Menu
6	Allocation	Allocation Support Menu
3	Channel Gain Setting Option	Auto Gain Prompt Channel 1
N	Auto Gain Prompt	Auto Gain Prompt Channel 2
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 3
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 4
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 5
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 6
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Allocation Support Menu
4	GDA Pointing Option	Option Select Prompt
ON	Turn Pointing ON	Allocation Support Menu
5	Downlink MBX Allocation Option	Option Select Prompt
OFF	Turn Option Off	MBX Option Menu
Level-Up	Return To Allocation Menu	
6	Uplink MBR Allocation Option	Option Select Prompt
OFF	Turn Oprtion Off	MBR Option Menu
Level-Up	Return To Allocation Menu	
7	Frequency Hop ALlocation Option	Option Select Prompt
OFF	Turn Option Off	Allocation Menu
1	Initiate Network Allocation Processing	GDA Pointing Algorithm
Level-Up	ECCM Allocation Modification Menu	Allocation Menu

1	MBR Modifications	MBR Modifications Menu
1	Jammer Nulling Algorithm	Jammer Constraints Definitions
Level-Up		MBR Modifications Menu
Level-Up		Modifications Menu
Level-Up		Allocation Menu
6	Uplink MBR Allocation Option	Option Prompt
OFF	Turn Option Off	MBR Option Menu
1	Beamweights For MBR	MBR Beamweight Prompt
E	Select ECCM Beamweights	MBR Option Menu
Level-Up	Return To Allocation Menu	
1	Initiate Allocation Processing	GDA Pointing Algorithm
Level-Up		Modification Menu
Level-Up		Allocation Menu
Level-Up		Top Level DNPS Menu
7	Performance	Performance Support Menu
5	ACI Effects Option	Option Select Prompt
OFF	Turn Option Off	Performance Support Menu
6	Intermodulation Option	Option Select Prompt
OFF	turn Option Off	Performance Support Menu
1	Initiate Perf Analysis W/O Margin Leveling	Perf Menu
Level-Up	Top Level DNPS Menu	
11	Utilities	Utilities Menu
3	ECCM Adaption	Adaption Subsystem Menu
10	Modify Alloc/Perf Processing Flags	Alloc/Perf Options
1	Channel Gain Settings Option	Auto Gain Setting Prompt
N	Select Auto Gain Channel 1	Channel 2 Prompt
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 3
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 4
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 5
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 6
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Alloc/Perf Options Menu
3	Downlink MBX Allocation Option	Option Select Prompt
OFF	Turn Option Off	MBX Option Menu
Level-Up		Alloc/Perf Options Menu
4	Uplink MBR Allocation Option	Option Select Prompt
OFF	Turn Option Off	MBR Option Menu
1	Beamweight For MBR	Beamweight Select Prompt
E	Use Eccm Beamweights	MBR Option Menu
Level-Up		Alloc/Perf Options Menu
5	Frequency Hop Allocation Option	Option Select Prompt
OFF	Turn Option off	Alloc/Perf Optios Menu
6	Jammer Nulling Option	Option Select Prompt
ON	Turn Option Off	Alloc/Perf Option Menu
Level-Up		ECCM Top Level Menu
13	Select/Modify Interrupt Criteria	Criteria Menu
3	Step Through Traffic Elements	Number Of Steps Prompt
1	Select 1 Element	Criteria Menu
Level-Up	ECCM Top Level Menu	
15	Execute Adaption	E42 Adaption In Progress Message
		E46 Adaption Breakpoint Encountered
17	Traffic Ranking Report	
15	Execute Adaption	E42 Adaption In Progress Message
		E46 Adaption Breakpoint Encountered
17	Traffic Ranking Report	
13	Select/Modify Interrupt Criteria	Criteria Menu
3	Step Through Traffic Elements	Number Of Steps Prompt
9	Select 9 Elements	Criteria Menu
Level-Up	ECCM Top Level Menu	
15	Execute Adaption	E42 Adaption In Progress Message
		E46 Adaption Breakpoint Encountered
17	Traffic Ranking Report	
13	Select/Modify Interrupt Criteria	Criteria Menu
1	Entire Network	
Level-Up	ECCM Top Level Menu	
15		E42 Adaption In Progress Message
		E43 Adaption Complete
17	Traffic Ranking Report	
14	Report Generation	Report Generation Menu
1	Network Reports	Network Reports Menu
1	Network Performance Summary	Output Prompt

P	Output To Printer	Network Reports Menu
Level-Up	Report Generation Menu	
2	Link Reports	Link Reports Menu
2	Link Performance Summary (For A Channel)	Output Prompt
P	Output To Printer	Channel Select Prompt
1	Select Channel 1	Channel Select Prompt
2	Select Channel 2	Channel Select Prompt
3	Select Channel 3	Channel Select Prompt
4	Select Channel 4	Channel Select Prompt
Level-Up	To Link Reports Menu	Link Reports Menu
Level-Up	Report Generation Menu	
3	Terminal Reports	Terminal Report Menu
2	Terminal/Jammer Dir Gains Summary	Output Prompt
P	Output To Printer	Terminal Report Menu
Level-Up		Report Generation Menu
7	Satellite Analysis Summary	Output Prompt
P	Output To Printer	Report Generation Menu
9	ECCM Allocation Summary	Output Prompt
P	Output To Printer	Reprot Generation Menu
14	MBX Beamweights	MBX Options Menu .
1	Display Allocation Beamweights	MBX Options Menu
Level-Up		Report Generation Menu
15	MBR Beamweights	MBR Options Menu
2	Display ECCM Beamweights	MBR Options Menu
Level-Up		Report Generation Menu
13	ECCM Reports	ECCM Report Menu
1	ECCM Circuit Parm Summary	Circuit Parm Summary
1	For All Circuits	Output Prompt
P	Output To Printer	Circuit Parm Summary Menu
Level-Up		ECCM Reports Menu
2	ECCM Circuit Adaption Criteria	Circuit Adaption Criteria Summary
1	For All Circuits	Output Prompt
P	Output To Printer	Circuit Adaption Criteria Summary
Level-Up		ECCM Reports
Level-Up		Report Generation Menu
Level-Up		ECCM Adaption Top Level Menu
Level-Up		Utilities Menu
Level-Up	Top Level DNPS Menu	Top Level Menu
13	Logout Of DNPS	Previously Saved Scenarios
Level-Up	Confirms Logout	Top Level DNPS Menu
13	Logout	RAS Logout Message

A.4.3 Scenario Analysis and Conclusions

Due to the computer hardware problems at the DCEC Reston facility, results of this effort were delayed. Results will be reported under follow-on tasking.

A.5 Test Scenario Two

A.5.1 Scenario Description

Scenario two included definitions for ECCM links and FDMA links. Many different user groups were defined, and prioritization of these groups varied from '000000' to '999998'. Criteria definitions of this scenario provided for the software's ability to select links of the network based on the criteria selected. The criteria defined within the scenario were for geographic location, user classification, data rates and communications classification. Link definitions in the scenario provided for traffic across channels 1, 2, 3 and 4.

A.5.2 Scenario Two Test Steps

Input	Description	Output
Username	Enter username	Password prompt
Password	Enter password	DOSS/DNPS Top level menu
4	Scenario Definition	Scenario Definition Menu
9	Mutiple Scenarios	Catalog Of Saved Scenarios
2	Retrieve A Scenario	Prompt To Select A Scenario
6	Index Of Scenario	Options For Scenario Loading
8	Retrieve Entire Scenario	Database Reset Prompt
Y	Verifies Database Reset	Scenario Catalog Menu
Level-Up	Return To Scenario Definition Menu	Scenario Definition Menu
Level-Up	Return To Top Level DNPS Menu	Top Level DNPS Menu
5	Analysis Support	Analysis Support Menu
1	Initialize Satellite Constants	Satellite IRON Prompt
6451	The IRON Of The Satellite	Analysis Support Menu
4	List/Modify Satellite SSP	Satellite Subpoint Position Detail
4	Satellite Longitude	Logitude Prompt
-135	Enter Longitude For Satellite	Satellite Subpoint Position Detail
Level-Up	Return To Analysis Support Menu	Analysis Support Menu
5	Perform Satellite Analysis	Analysis Support Menu
Level-Up	Return To Top Level DNPS Menu	Top Level DNPS Menu
6	Allocation	Allocation Support Menu
3	Channel Gain Setting Option	Auto Gain Prompt Channel 1
N	Auto Gain Prompt	Auto Gain Prompt Channel 2
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 3
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 4
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 5
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Auto Gain Prompt Channel 6
Y	Auto Gain Prompt	Option Select Prompt
OFF	Turns Auto Gain Off	Allocation Support Menu
4	GDA Pointing Option	Option Select Prompt
ON	Turn Pointing ON	Allocation Support Menu
5	Downlink MBX Allocation Option	Option Select Prompt
ON	Turn Option On	Allocation Support Menu
6	Uplink MBR Allocation Option	Option Select Prompt
On	Turn Option On	Allocation Support Menu
7	Frequency Hop ALlocation Option	Option Select Prompt
OFF	Turn Option Off	Allocation Menu
8	Summary Display Option	Option Prompt
ON	Turn Option On	Allocation Support Menu
1	Initiate Allocation Processing	GDA Point Algorithm
Level-Up		Allocation Support Menu
5	Downlink MBX Allocation Option	Option Prompt
OFF	Turn Option Off	MBX Option Menu
Level-Up		Net. Resource Allocation Menu
6	Uplink MBR Allocation Option	Option Prompt
OFF	Turn Option Off	MBR Options Menu
Level-Up		Net. Resource Allocation Menu
1	Initiate Net. Resource Allocation Processing	GDA Pointing Algorithm
Level-Up		Net. Resourc Allocation Menu
Level-Up		Top Level DNPS Menu
7	Performance	Performance Support Menu
5	ACI Effects Option	Option Select Prompt
OFF	Turn Option Off	Performance Support Menu
6	Intermodulation Option	Option Select Prompt
OFF	turn Option Off	Performance Support Menu
1	Initiate Perf Analysis W/O Margin Leveling	Perf Menu
Level-Up	Top Level DNPS Menu	
11	Utilities	Utilities Menu
3	ECCM Adaption	Adaption Subsystem Menu
10	Modify Alloc/Perf Processing Flags	Alloc/Perf Options
1	Channel Gain Settings Option	Auto Gain Setting Prompt
N	Select Auto Gain Channel 1	Channel 2 Prompt
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 3
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 4
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 5

Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Auto Gain Setting Prompt Channel 6
Y	Modify Auto Gain	Auto Gain Prompt
OFF	Turn Option Off	Alloc/Perf Options Menu
3	Downlink MBX Allocation Option	Option Select Prompt
OFF	Turn Option Off	MBX Option Menu
Level-Up		Alloc/Perf Options Menu
4	Uplink MBR Allocation Option	Option Select Prompt
OFF	Turn Option Off	MBR Option Menu
Level-Up		Alloc/Perf. Options
5	Frequency Hop Allocation Option	Option Prompt
OFF	Turn Option Off	Alloc/Perf. Options Menu
Level-Up		ECCM Top Level Menu
13	Select/Modify Interrupt Criteria	Criteria Menu
3	Step Through Traffic Elements	Number Of Steps Prompt
1	Select 1 Element	Criteria Menu
Level-Up	ECCM Top Level Menu	
15	Execute Adaption	E42 Adaption In Progress Message E46 Adaption Breakpoint Encountered
17	Traffic Ranking Report	
15	Execute Adaption	E42 Adaption In Progress Message E46 Adaption Breakpoint Encountered
17	Traffic Ranking Report	
13	Select/Modify Interrupt Criteria	Criteria Menu
3	Step Through Traffic Elements	Number Of Steps Prompt
9	Select 9 Elements	Criteria Menu
Level-Up	ECCM Top Level Menu	
15	Execute Adaption	E42 Adaption In Progress Message E46 Adaption Breakpoint Encountered
17	Traffic Ranking Report	
13	Select/Modify Interrupt Criteria	Criteria Menu
1	Entire Network	
Level-Up	ECCM Top Level Menu	
15		E42 Adaption In Progress Message E43 Adaption Complete
17	Traffic Ranking Report	
14	Report Generation	Report Generation Menu
1	Network Reports	Network Reports Menu
1	Network Performance Summary	Output Prompt
P	Output To Printer	Network Reports Menu
Level-Up	Report Generation Menu	
2	Link Reports	Link Reports Menu
2	Link Performance Summary (For A Channel)	Output Prompt
P	Output To Printer	Channel Select Prompt
1	Select Channel 1	Channel Select Prompt
2	Select Channel 2	Channel Select Prompt
3	Select Channel 3	Channel Select Prompt
4	Select Channel 4	Channel Select Prompt
Level-Up	To Link Reports Menu	Link Reports Menu
Level-Up	Report Generation Menu	
3	Terminal Reports	Terminal Report Menu
2	Terminal/Jammer Dir Gains Summary	Output Prompt
P	Output To Printer	Terminal Report Menu
Level-Up		Report Generation Menu
7	Satellite Analysis Summary	Output Prompt
P	Output To Printer	Report Generation Menu
9	ECCM Allocation Summary	Output Prompt
P	Output To Printer	Reprot Generation Menu
14	MBX Beamweights	MBX Options Menu
1	Display Allocation Beamweights	MBX Options Menu
Level-Up		Report Generation Menu
15	MBR Beamweights	MBR Options Menu
2	Display ECCM Beamweights	MBR Options Menu
Level-Up		Report Generation Menu
13	ECCM Reports	ECCM Report Menu
1	ECCM Circuit Parm Summary	Circuit Parm Summary
1	For All Circuits	Output Prompt
P	Output To Printer	Circuit Parm Summary Menu
Level-Up		ECCM Reports Menu
2	ECCM Circuit Adaption Criteria	Circuit Adaption Criteria Summary
1	For All Circuits	Output Prompt
P	Output To Printer	Circuit Adaption Criteria Summary
Level-Up		ECCM Reports
Level-Up		Report Generation Menu
Level-Up		ECCM Adaption Top Level Menu

Level-Up		Utilities Menu
Level-Up	To. .evel DNPS Menu	Top Level Menu
13	Logout Of DNPS	Previously Saved Scenarios
Level-Up	Confirms Logout	Top Level DNPS Menu
13	Logout	RAS Logout Message

A.5.3 Scenario Analysis and Conclusions

Due to computer hardware problems at the DCEC Reston facility, results of this effort were delayed. Results will be reported under follow-on tasking.

A.6 User Test Scenario

A.6.1 Scenario Description

The user scenario provides for verification of output of different versions of DOSS/DNPS software. This scenario was used to create output from the various subsystems of DNPS of two versions of software. This output is used for verification of output continuity/integrity between versions. The scenario calls for manipulation of data by the software using the same set of commands that the author normally uses. This scenario was established as a feasibility study of traffic in channel 1. Although the scenario contains definitions for links in other channels, the scenario was primarily used for the evaluation of channel 1 traffic.

A.6.2 User Scenario Test Steps

Input	Description	Output
Username	Enter username	Password prompt
Password	Enter password	DOSS/DNPS Top level menu
4	Scenario Definition	Scenario definition menu
9	Mutiple Scenarios	List of saved scenarios
7	Load scenario	Prompt for scenario number
8	Load entire scenario	Entire scenario is loaded
Y	Reset active database	Scenario is loaded, database is reset
Level-up	Scenario Definition	Scenario definition menu
Level_up	Doss/DNPS Top level	Top Level Menu
5	Analysis Support	Analysis Support Menu
1	Initialize Satellite Constants	Satellite IRON Prompt
6451	Satellite IRON	Analysis Support Menu
4	List Modify Satellite SSP	Satellite Subpoint Position Detail
4	Satellite Longitude	Longitude Prompt
-12	Desired Longitude	Satellite Subpoint Position Detail
Level-up		Analysis Support
5	Perform Satellite Analysis Processing	Analysis Support Subsystem
Level-up		DNPS Main Menu
6	Allocation	Allocation Menu
4	GDA Pointing Option	Option Select Prompt
OFF	Set to off	Manual GDA Point Menu
Level-up		Allocation Menu
5	Downlink MBX Allocation Option	Option Prompt
OFF	Set to off	MBX Options Menu
1	Beam Weights For M1X	Beamweight Set Prompt
S	Set to Scenario	MBX Option Menu

2	Beam Weights For M2X	Beamweight Set Prompt
S	Set To Scenario	MBX Option Menu
Level-up		Allocation Menu
6	Uplink MBR Allocation Option	Option Prompt
OFF	Set To off	MBR Options Menu
1	Beam Weights For MBR	Beamweight Set Prompt
S	Set To Scenario	MBR Option Menu
Level-up		Allocation Menu
1	Initiate Allocation Processing	Downlink MBX Allocation Menu
Level-up		Modifications Menu
Level-up		Allocation Menu
Level-up		Top Level Menu
7	Performance	Net. Performance Subsystem
8	Perf. Summary Option	Option Prompt
ON	Turn Option On	Net. Performance Subsystem
2	Initiate Perf. With Margin Leveling	Margin Leveling Menu
4	All Links	Margin Leveling Menu
2	All Channels	Net. Performance Subsystem
2	Initiate Perf. With Margin Leveling	Margin Leveling Menu
2	SSMA Links	Margin Leveling Menu
1	Select A Channel	Channel Select Prompt
1	Enter Channel 1	Net. Performance Subsystem
2	Initiate Perf. With Margin Leveling	Margin Leveling Menu
2	SSMA Links	Margin Leveling Menu
1	Select A Channel	Channel Select Prompt
1	Enter Channel 1	Net. Performance Subsystem
Level-up		Top Level DNPS
8	Report Generation	Report Generation Subsystem
1	Network Reports	Network Reports Menu
1	Network Performance Summary	Output Prompt
P	Output To Printer	Network Reports Menu
Level-up		Report Generation Subsystem
2	Link Reports	Link Reports Menu
1	Link Performance Summary (all links)	Output Prompt
P	Output To Printer	Link Reports Menu
Level-up		Report Generation Subsystem
3	Terminal Reports	Terminal Reports Menu
2	Terminal/Jammer Directive Gains Summary	Output Prompt
P	Output To Printer	Jammer Reports Menu
Level-up		Report Generation Subsystem
4	Jammer Reports	Jammer Reports Menu
2	Terminal/Jammer Directive Gains Summary	Output Prompt
P	Output To Printer	Jammer Reports Menu
Level-up		Report Generation Subsystem
5	MBA Plotting	MBA Plotting Menu
4	Contour Plot	COntour Change Prompt
N	No Change In Contour Levels	Land Contour Prompt
N	No Land Contour	MBA Plotting Menu
Level-up		Report Generation Subsystem
7	Satellite Analysis Summary	Output Prompt
P	Output To Printer	Report Generation Subsystem
Level-up		Top Level DNPS
13	Logout	Scenarios Are Displayed
Level-up		Top Level DNPS
13		Logout Complete

A.6.3 Scenario Analysis

Results of this effort will be reported under the FY86 contract tasking.

A.6.4 Conclusions

Results of this effort will be reported under the FY86 contract tasking.

A.7 DOSS/DNPS Version 3.1 Problems and Conclusions and Recommendations

A.7.1 Problems

The following are the problems encountered during the testing effort. These problems ranged from cosmetic errors to software logic problems. (Additionally, significant hardware problems/downtime was experienced during the tests; however, these problems are expected to be corrected in the near term.)

- **Documentation -**
  There is insufficient documentation of the ECCM Adaption subsystem. Although it is a very powerful addition to the DNPS package, more complete documentation of this subsystem is recommended.

- **SSMA Modem Table -**
  When a modem name is added and/or modified, the name must adhere to a strict naming convention. This naming convention is critical during the automatic creation of ECCM links that support a newly added ECCM terminal. One of the following is recommended:

  - The naming convention be documented (e.g., AN/USC-28 vs. USC-28)

  - The software be modified so that the modem name is not used with *program logic.*

  - The software parse the modem name during input. If a modem name is critical in subsequent program logic, then the software should test (parse) the input for a valid name. By preventing incorrect input for a name field, software logic failures will be eliminated. In this case, "AN/USC 28" or "AN/USC/28" would not satisfy the name required (AN/USC-28) *in subsequent software logic* (although *similar* to the required name).

- **Local Loop -**
  Two local loops were unable to meet predicted margins required to support the downlink. By switching the dominant link to RL1XXX (from RN1XXX) the local loop was included in the network.

- **Insufficient Testing of the Network Control Book -**
  Menu errors occurred during the examination of the Network Configuration Book (NCB).

- **Output Generation Message** -
The report generation message is issued erroneously when a Level-up is commanded to abort command. The Level-up command is issued at the output prompt (D/P) after selecting a specific report type. This message was noted from the ECCM Report Generation subsystem under the following report types:

  - o Terminal/Jammer Directive Gains Summary
  - o ECCM Allocation Summary.

A.7.2 Conclusions and Recommendations

The software needs more rigorous testing prior to being formally accepted. With the inclusion of the ECCM subsystem, major areas of the software were only preliminarily evaluated/tested. The following would aid in the testing effort:

- Determine the Extent of Testing -
When a Software User Report is accepted by the Configuration Control Board, the scope of testing should then be established.

- Review Code Prior to Acceptance Testing -
Desk check the code to determine the impact on other areas of code. This should be done prior to starting any testing. The results of the review will determine what sections need to be exercised.

- Establish Minimal Acceptance Test -
A minimal set of tests should be established and used for all versions of the software prior to release.

- Establish Automated Acceptance Tests -
Through the use of automated test procedures, the software can be tested by using a specific set of 'canned' tests. This will prevent the introduction of new problems.

- **Breakpoint Type/Number** -
During the generation of reports using the Adaption Report Generation subsystem, the event that caused the breakpoint to be achieved should be listed on the output. This could be the current adaption step or a message indicating that adaption had completed.

Appendix B

DNPS Overview And MBR_AAS Evaluation

PREVIOUS PAGE IS BLANK

B.1 SYSTEM OVERVIEW OF DNPS

The DNPS is used by the Defense Communications Agency (DCA) to plan for the allocation of resources of the Defense Satellite Communications System (DSCS) in support of communication requirements. The purpose of this package is to evaluate performance of various resource allocation strategies of the DSCS network. Such strategies are defined by user scenarios that model DSCS communication links, networks, satellites, earth terminals, and jammers. DNPS is the resource allocation/performance function of the two main functions that comprise the DOSS. The DSCS Automatic Spectrum Analyzer (DASA) is the companion DOSS function that allows monitoring of a DNPS-analyzed network. Typical DASA monitoring functions measure and/or compute power, received signal-to-thermal noise ratio (C/kT), and power versus frequency spectrum data for the entire transponder bandwidth or selected channels or links within a transponder. Such information is used for comparison against the results predicted by the DNPS model. DOSS/DASA controls reports and/or alarms that indicate equipment degradation or changes to network utilization. Together, DNPS and DASA functions enable a user to configure and monitor DSCS network configurations.

B.2 FUNCTIONAL DESCRIPTION OF THE DNPS

The DNPS implements several functions that support definition, analysis, and performance reporting of the satellite communication networks based on allocation of communication subsystem resources. Figure B-1 depicts the functional organization of DNPS. A summary of the functions follows:

- **DNPS Supervision**
  The DNPS supervisor provides operator control over all DNPS functions via menus and prompts to the operator. In addition, it provides communications with DASA, 24-hour activity logging of users, and maintenance of system state for function initialization.

- **Procedure Definitions**
  Procedure definition allows the operator to store a sequence of keystrokes that may be later retrieved and executed in place of operator command input. If a procedure file has an illegal keystroke sequence, the file processing is

Figure B-1. Functional Organization of DNPS

```
                +----------------------+
                |                      |
                |  DNPS   Supervisor   |
                |                      |
                +----------+-----------+
                           |
     +---------------------+
     |
     +---------+----------------+--------------+-------------+
     |         |                |              |             |
     |  +------v------+ +-------v------+ +------v------+ +-----v------+
     |  |  Procedure  | | Static Data  | |  Scenario   | |  Analysis  |
     |  | Definitions | | Base Loading | | Definitions | |  Support   |
     |  +-------------+ +--------------+ +-------------+ +------------+
     |
     +---------+----------------+--------------+-------------+
     |         |                |              |             |
     |  +------v------+ +-------v------+ +------v------+ +-----v------+
     |  | Satellite   | | Net. Resource| |Net. Perform.| |   Report   |
     |  |  Loading    | |  Allocation  | |  Analysis   | | Generation |
     |  +-------------+ +--------------+ +-------------+ +------------+
     |
     +---------+----------------+--------------+-------------+
               |                |              |             |
        +------v------+ +-------v------+ +------v------+ +-----v------+
        |    ECCM     | |     DASA     | | Sat. Tasking| |  Special   |
        |  Adaptation | |              | |  & Config.  | |Requirements|
        +-------------+ +--------------+ +-------------+ +------------+
```

terminated automatically. Command input is resumed from the user's console.

- **Static Data Base Loading**
Automatic loading of the static data base is performed by this function. It loads all files into memory that are necessary to support DNPS functions. If a privileged user (operational account) changes these data, then the static data base loader ensures that all inputs are within bounds.

- **Scenario Definition**
Ten scenario definitions are accessible through a user's library or a copy function from another user's account. The scenario definition function allows a user to select from the above and/or build a new scenario based on the active scenario. This is accomplished by selections, additions, deletions, or modifications to the dynamic data base (the active scenario). Such changes to the active scenario are checked for legitimate entry, and errors are reported. Access to a stored data base is checked against the static data base for compatibility. The operator is warned of all incompatible elements.

- **Analysis Support**
The analysis support function is activated for an operator-defined scenario in order to generate needed information for the resource allocation and performance analysis or report functions. Several calculations are performed based on the active scenario:

  - o Ephemeris generation from the orbit vector.
  - o Terminal/jammer range, range/rate.
  - o Terminal/jammer azimuth/elevation locations (with satellite-in-view or out-of-view information).
  - o Calculations for normal atmospheric, free space, and total path loss.
  - o A terminal/jammer MBA pattern index generator used for accessing singlet data on a sorted highest to lowest beam contribution amplitude. This is done for each terminal and jammer for each MBA.
  - o Calculation of the gimbal dish antenna (GDA) directive gain based on pointing information.
  - o Earth coverage directive gain for all antennas.

- **Satellite Loading**
The satellite loading function provides automatic allocation of channel assignment, coding type, and modulation type for operator-specified FDMA links. Criteria for this function are based on available satellite power/bandwidth and constraints imposed on link, link modulation/coding, and antenna/channel connectivity defined by the active scenario.

• **Network Resource Allocation**
The network resource allocation function selects all network parameters required to compute network performance analyses. The operator can select parameters manually or call algorithms which generate them automatically. The following parameters are assigned or utilized:

- Uplink transmitter power and effective isotropic radiated power (EIRP) for each network link
- Transponder channel gain settings
- GDA pointing coordinates and directive gains
- Satellite transponder/channel connectivity
- Quantized beamweights and directional gains for all MBAs

• **Network Performance Analysis**
The network performance analysis function estimates network performance using analytic and tabular models for terminal, uplink, satellite, and downlink parameters and effects. Input consists of operator-selected options and network parameters determined by the resource allocation function. Outputs can be compared to measurable network parameters such as link margins, terminal power margins, received signal strength, C/kT transponder output backoff, etc. The operator uses the network performance analysis function to obtain and display the network performance.

• **Report Generation**
The report generation function is responsible for formatting and outputting reports to various soft and hardcopy devices. It is capable of accessing the static data base and active scenario data. The following accounting information is in each header:

- Date/time
- Security Classification
- Operator Identification
- Scenario Identification
- Title
- System State
- Page Number (e.g., 1 of 5)

Reports cover all facets of DNPS functions. They are embodied in many

report types. Report topics aggregate user-defined parameters, models, and configurations that comprise all DOSS capabilities. Topics are:

- o Network Performance and Configuration Summaries
- o Link Performance, Analysis, and Allocation
- o Terminal and Jammer Characteristics
- o Satellite Analysis and Parameters
- o Antenna Characteristics including Contour Plots
- o ECCM Network, Link, Terminal, and Parameter Summaries
- o DASA Alarms and Reports

- **ECCM Adaptation**
  The ECCM function sorts and assigns traffic loads to allocate as many circuits as possible in a stressed environment. Reduced traffic loading in a stressed environment must satisfy network performance margins, constraining parameters and traffic ranking definable for the DNPS III environment. Constraints ignore all links and terminals which are out-of-view or turned off. Power-constrained links are not altered.

- **DASA**
  The DASA function provides an interface between DNPS and the DASA system. It checks operator input for validity and transmits configuration and control commands to DASA. This function receives and formats reports and alarms from DASA.

- **Satellite Tasking and Configuration**
  The satellite tasking and configuration function provides the Man-Machine Interface (MMI) necessary for operator control of data transfer to and from the Satellite Configuration Control Element (SCCE). The SCCE is responsible for satellite monitoring telemetry, and configuration control of the satellite communication subsystem.

B.2.1 Operational Accounts

Operational accounts contain the operational scenario of the actual network configuration. As such, they are typically assigned to the Satcom Network Controller (SNC), who can exercise a broad range of capabilities. An operational account has five basic capabilities. The first three three items are contingent on DCA Operations Center (DCAOC) approval.

- Modify the DNPS online static data base. This includes static terminal, static jammer, modem, filter, multiplexer and satellite ephemeris data.

- Modify the operational scenario. This includes modification of communication link, satellite configuration, and earth terminal parameters.

- Create new DNPS data bases (new operational data) that can be passed to the DASA system for use in monitoring the downlink signal.

- DASA commands and acknowledgement of DASA alarms.

- Use of all privileges available to analyst accounts.

B.2.2 Analyst Accounts

Analyst accounts allow the user to plan scenarios that do not modify the operational scenario. Each analyst account user may interact with the DNPS software by creating or modifying scenarios from his/her library of scenarios. Scenario interaction is performed on the active scenario. Saved scenarios defined in analyst accounts may be copied into other analyst accounts.

B.3 SYSTEM ARCHITECTURE

The DNPS system architecture may be examined with respect to the software and hardware configurations as discussed below.

B.3.1 Software Configuration of DNPS

DNPS is configured as a single shared image with multiuser access. As a single task it supports multiple users by multiprocessing. Access to DNPS is granted by limiting execution to users with valid operational or analyst accounts defined on a site basis. A login procedure file checks the validity of the user's account prior to program execution. DNPS is menu driven, with submenu selections displayed upon entry to the function. User consoles are formatted with the current function and a prompt for selectable menu items or parameter input for a specific function. Regression from submenus is accomplished by definition of the "level-up" function key. This key cancels the current function level and returns to the next highest level of menus. In this way the operator can control the sequence of all program execution.

B.3.2 Hardware Configuration of DNPS (DCEC and Field Sites)

Hardware configuration of DCEC and field site systems varies. Field sites have been or are being configured with Digital Equipment Corporation VAX 11/730 front ends to replace the PDP-11/04 as shown below in Figure B-2. The DCEC configuration supports an additional man-machine interface that is not on the field equipment. This description applies to the DCEC configuration with field site exceptions noted. The major component of DNPS is a VAX-11/780 connected to one of six user stations. Two remote access stations consist of a PDP-11/04 connected to the VAX via DECnet. The PDP-11/04s support their Tektronix 4014 graphics terminal, Versatec 1200A printer/plotter and the Televideo 920 C console. (Twin structures are noted as <1 of 2>). The local access configuration (DCEC only) supports a Tektronix 4014 console, a Versatec 1200A printer/plotter, and five VT 100 user consoles. DCMTEST (a software module) emulates the PDP-11/04 function for the local access station and performs input data conversion from the VT100 terminal.

B.3.3 Man-Machine Interface

The man-machine interface (MMI) is the basic component that provides user input/output access. There are four peripherals that give the user physical access to DNPS. They allow the user to select from a menu (or submenu), enter parameter values, display parameters/reports, and make hardcopies of parameters/reports. They are:

- **VT100 Terminal** Primary menu input device for local commanding to DNPS software at DCEC. **TVI 920 A Terminal** Primary menu input device for remote commanding of DNPS. **Tektronix 4014** Primary report device for displaying system parameters or report generation output. Secondary features include the X-Y cursor control knobs which are used in defining area/gain coutour definitions for the MBR_AAS. (This device is not supported for input at remote sites.)

- **Versatec Printer/Plotter** Primary hardcopy device that directly copies the Tektronix CRT information by manual command.

- **Tektronix 4631 Hardcopy Device** Copies graphic display output from the Tektronix 4014 Display

Figure B-2. System Overview of Hardware Configuration (DNPS)

```
               +---------------------------------+
               |                                 |
               |       V A X - 11/780  CPU       |
               |                                 |
               +---+-----------+-----------------+
                   |<1 of 2>   |
          +--------+-------+ +-+------------------+
     +---+ PDP-11/04 CPU   | | DCMTEST (software) |
     |    +----------------+ +-+------------------+
     |                         |
     |    +----------------+   |    +------------------+
     |    |   TVI - 920C   |   |    |      VT100       |
     +---+                 |   +---+     (1 of 5)      |
     |    |    Console     |   |    |     Console      |
     |    +----------------+   |    +------------------+
     |                         |
     |    +----------------+   |    +------------------+
     |    |   Tektronix    |   |    |   Tektronix      |
     +---+                 |   +---+                   |
     |    |  4014 Display  |   |    |  4014 Display    |
     |    +----------------+   |    +------------------+
     |                         |
     |    +----------------+   |    +------------------+
     |    | Versatec 1200 A|   |    |   Tektronix      |
     +---+                 |   +---+                   |
          |Printer/plotter |        |  4631 Hardcopy   |
          +----------------+        +------------------+
```

B.4 PROBLEMS AND RECOMMENDATIONS

While the DNPS system has extensive capabilities, a major issue is centered on its single-task nonoverlaid environment. The execution module includes roughly 450 subprograms. These subprograms support 17 functions, which manipulate or display common data areas. Recommendations for DNPS system configuration are described below:

- **Use of Overlays**
  Overlay loading is a VAX/VMS linkage option that allows subprograms to share a common memory space. This technique reduces memory requirements on the VAX computer as well as VAX system requirements for each user.

- **Use of Multitasking**
  Multitasking may be used to separate functional elements of DNPS into independent tasks. As multitasks, each function is compiled, linked, and executed as a separate task. Users in the program development cycle would not be required to embody the entire DNPS in their work effort. DNPS would be relieved of evolutionary strain in that new or modified tasks would not compete for space requirements imposed by a large single task.

B.5 RESULTS OF THE MBR_AAS EVALUATION

The MBR_AAS is a working version of the DCEC software that implements the algorithm described in the subtask C statement of work. The modules TESTTEK, TEKSORT, and RETMBR, which make up the MBR_AAS, operate as three stand-alone packages. Integration of MBR_AAS requires an understanding of the methods used by the MBR_AAS and DNPS to generate or utilize MBR beam weights. This report describes methods for steering criterion of the MBR, summarizes DNPS methods of beam weight selection, and details the data and algorithms used by the MBR_AAS. Case studies of the MBR_AAS are summarized with conclusions on MBR_AAS performance.

B.6 SUMMARY OF METHODS USED FOR MBR STEERING ALGORITHMS

Methods used to produce MBR beam weights depend on the function used to generate them. The DNPS employs three different methods that generate separate MBR beam weights. They are summarized by the following:

- **Network Resource Allocation**
  Network resource allocation defines a set of beam weights based on terminal loading.

- **ECCM Adaptation**
  ECCM adaptation generates beam weights using the same algorithm as Resource Allocation. The algorithm is iterative and based on a stressed environment.

- **Scenario Definition**
  Scenario definition allows the operator to manually input beam weights.

Each method of input generates a set of beam weights. While the resource allocation and ECCM adaptation beam weights are interdependent, the final selection of active scenario beam weights may be chosen by the operator. Report results are generated by computing the directive gain for the entire azimuth/elevation field of view based on active scenario beam weights. (Directive gain is defined later in this appendix.) Plotting routines map the directive gain levels into azimuth/elevation coordinates. A template of the earth with continental boundaries can be overlaid on this gain contour. The gain contour process is then complete.

The MBR_AAS algorithm was developed to automate the contouring of directive gains by defining global areas that have high or low sensitivity (gain). Such areas can represent locations for friendly or hostile earth terminals. The MBR_AAS will take the defined areas as input and generate a set of complex beam weights that attempt to satisfy the area/gain definition. The MBR_AAS methodology consists of three stand-alone packages. TESTTEK defines area/gain constraints. TEKSORT sorts these constraints on an area basis. RETMBR computes beam weights that attempt to satisfy these constraints. Graphics software is included that allows various report generation capabilities. The following sections highlight the important features of the three packages.

B.6.1 TESTTEK

TESTTEK provides a means to define unique areas overlaid on a global map with continental boundaries. Several areas may be encompassed with nonintersecting polygons. TESTTEK has two major functions: providing the man-machine interface (MMI) and generating an output file that stores the selections made in this section. Input consists of map area/gain definitions. Primary inputs are user-defined polygons around areas of interest. They are made by X/Y knob positioning of cross hair cursors of the Tektronix 4014. Such areas are labelled (three characters) and assigned minimum/maximum gain constraints. Up to 30 areas may be defined.[2] When the data input phase is complete, a file is written to reflect the operator's area/gain input session.

B.6.2 TEKSORT

TEKSORT is an execution module called directly after TESTTEK. It has three main subprograms that perform area priority sorting. Areas of interest are defined by the TESTTEK output file. They are described in terms of azimuth/elevation locations, area labels, and gain constraints. These data are sorted on an area size basis so the steering algorithms give an equal priority to all areas under consideration proportional to their size. An output file is generated that contains the sorted results. Section B.7.6 details this method.

B.6.3 RETMBR

RETMBR is the program that embodies the beam forming process. This package takes the sorted file from TEKSORT and applies the beam forming algorithms to match the area definitions defined in TESTTEK. RETMBR also contains various display/report features that summarize intermediate results and writes a beam weight file for final results. The program has two basic outcomes; it will converge to the constraint requirements or report an impasse condition if the area/gain definition has not been realized. The goal of this program is to automate the beam weight selection process so that the operator skills necessary to produce the required operational gain contours are reduced.

[2]The number of areas is a compile time constant.

B.7 SUMMARY OF METHODS DATA AND ALGORITHMS USED BY RETMBR

The following sections are related to the RETMBR and are discussed in order to provide an understanding of the linear algebra and numerical analysis associated with MBR steering algorithms used in RETMBR.

B.7.1 Linear Algebra Used in MBR Algorithms

This section details a basic steering model, the linear algebra and the assumptions used in the directive gain and steering algorithms found in RETMBR. The RETMBR module is used to develop and display beam weights. Area/gain parameters are defined in the input phase of TESTTEK. The area of interest corresponds to the singlet data matrix (**A**), and corresponding gain limits for these areas are defined in the **Boundhi/low** vectors. These input parameters are used to produce optimization variables of the RETMBR package. Five basic variables are defined and used in the computational modules of RETMBR. The majority of these variables reside in RETMBR subprograms prefixed with "OPT" {**OPT000 through OPT351**}. They are:

- **X** - The normalized beam weight vector
- λ - The directive gain vector
- **PD** - The partial derivative of directive gain with respect to beam weights
- **Tconst** - The constraint equation
- **BC** - The beam quality equation

Two variables are defined for complex numbers within RETMBR. They represent identical operations for the real and imaginary parts of the variable. The following sections use complex notation, so these definitions are used to simplify complex operations.

B.7.2 Cost Function and Steepest Descent Algorithms for a Multiple Beam Antenna

All MBR antenna beam steering algorithms must have a performance criterion or cost function for optimization of beam weights. This section reviews some basic techniques of adaptive beam forming. A classical cost function is the minimization of the mean squared error between the actual gain contour and the desired gain contour. This penalty is used to avoid large deviations between the desired and actual beam weights. The cost function has the form:

$$CF = \sum_{area} (gain_{terminal} - gain_{desired})^2 \Rightarrow 0 \qquad B.1$$

where:

CF	Cost function; the parameter to minimize
terminal	An individual terminal
area	All the area locations
$gain_{terminal}$	Computed receive antenna gain for earth location terminal
$gain_{desired}$	Desired receive antenna gain for earth location terminal
=>	Minimize the function to approach

The steepest descent steering algorithm employs the gradient operator to update beam weight selection. This algorithm updates beam weights by selecting them in the *negative gradient* of the cost function. This is represented by:

$$W(k+1) = W(k) - \mu \nabla(CF) \qquad B.2$$

where:

W(k)	Current beam weights (k^{th} iteration)
W(k+1)	Next set of beam weights ($[k+1]^{th}$ iteration)
μ	A bounded constant selected to make the algorithm converge
∇	The gradient operator:

$$\nabla(CF) = \partial(CF)/\partial Re(X) + \partial(CF)/\partial Im(X)$$

where:

X	The complex beam weight space: $W = Re(X) + jIm(X)$
CF	The cost function
Re(),Im()	The real and imaginary parts of the vector

Two factors that affect the steepest descent (negative gradient) method are the selection of convergence constant (μ) and quantization of the beam weights imposed by DCSC III satellite electronics. The convergence constant is a scaling factor. It scales how much of the gradient should be used to adjust the beam weights. By choosing the scale factor too large, the algorithm will diverge or oscillate. However, a small scale factor makes the algorithm converge too slowly. A second factor affecting performance is the satellite implementation of power and phase states. The DSCS III MBR antenna feed

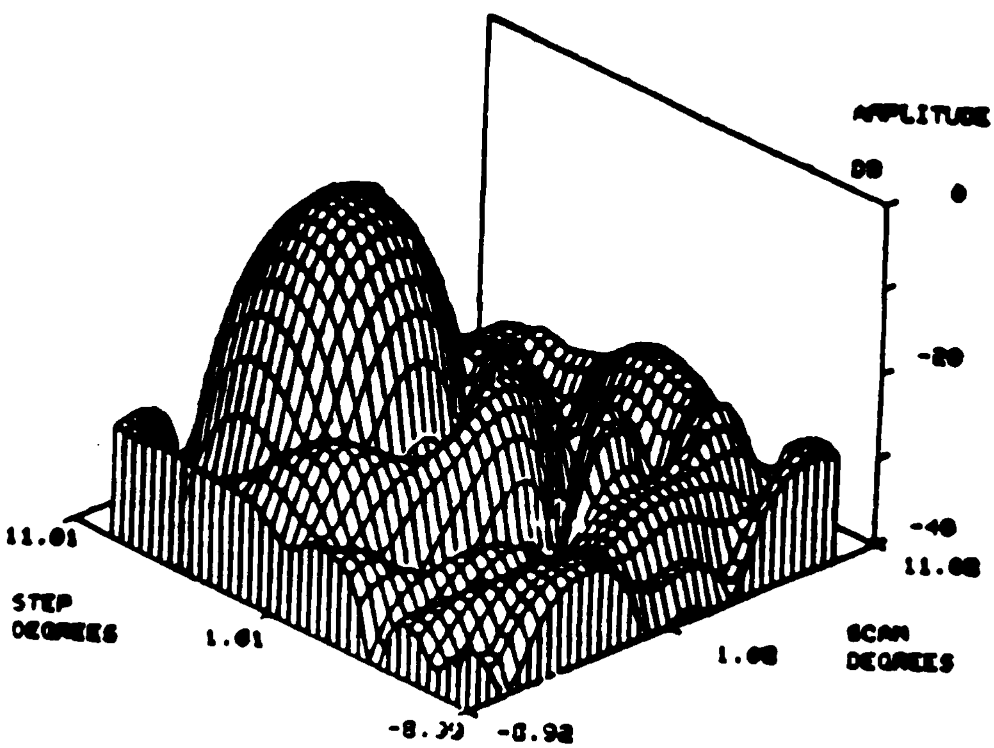
electronics produce discrete phase shifts and power levels. The gradient algorithm must be consistent with the limited number of discrete values possible in the actual equipment.

To cope with these design problems, DCEC Code R420 developed an algorithm to account for these design problems. This algorithm accounts for discrete step size limitations in the power/phase electronics that are inherent in the satellite. The model further employs an area-selection algorithm that gives equal priority to all areas of interest. Methods used in this algorithm are described in the following sections.

B.7.3 Singlet Data

Singlet data is the electromagnetic field strength that each singlet beam can realize at a given azimuth/elevation coordinate (0.2° granularity). Although these singlet beams should ideally be circularly symmetric with $J_1(x)/x$ gain, a review of individual singlet beam contours reveals significant asymmetries. Figure B-1 shows a typical singlet beam pattern for ±9° azimuth/elevation angles that encompass earth coverage. The non-symmetric gain sidelobes are a result of lens aberrations and other factors.

Figure B-3. Typical Singlet Beam Pattern for MBR



The selected singlet data matrix (**A**) is defined as the subset of singlet data for the areas of interest. These areas are operator-defined in the TESTTEK package. The array **A** has the form:

<u>Selective Singlet Data or Beam Contribution Array</u> {M,61}

$$\mathbf{A} = \begin{vmatrix} \alpha_{1,1}+j\beta_{1,1} & \cdots & \alpha_{1,61}+j\beta_{1,61} \\ \alpha_{2,1}+j\beta_{2,1} & \cdots & \alpha_{2,61}+j\beta_{2,61} \\ . & \cdots & . \\ . & \cdots & . \\ \alpha_{M,1}+j\beta_{M,1} & \cdots & \alpha_{M,61}+j\beta_{M,61} \end{vmatrix} \qquad \textbf{B.3}$$

Each singlet element is a complex value that is defined in terms of real (α) and imaginary (β) parts. A general element ($\alpha_{i,j}$ + $\beta_{i,j}$) represents the j^{th} beam contribution at the i^{th} azimuth/elevation point selected from the field of view. Each row represents the beam vector for a given point and each column represents the area vector for each singlet beam. The number of columns is fixed at 61. The number of rows or areas (M) may vary from one point to full coverage of 8281 points (the 91 x 91 grid).

B.7.3.1 Area Bound Constraints or Desired Gain

The TESTTEK program defines the desired gain constraints for the areas of interest. An area of interest contains a set of points enclosed by a user-drawn polygon. All points within the polygon have the same desired gain. They are referred to as the area bound constraints. These constraints are specified in terms of high and low gain limits:

<u>Area Definition Bound Limits</u> {na x 2}

Boundhi(na) = User defined (dB) **B.4**

Boundlow(na) = User defined (dB) **B.5**

Where;

na is the number of user-defined areas

B.7.3.2 Beam Weight Vector

The beam weight vector (B) contains the initial beam weights that may vary from ±1 continuously. While the current MBR_AAS program uses a set of zeroed beam weights as the initial beam weight vector, there is no restriction on the beam weights. This implies that beam weights may be supplied from other user results or inputs. The vector shown is an example of the beam weight matrix with the maximum boundary constraints:

Beam Weights {61,1}

$$BW_i \;\pm 1.000;\; i=1,61 \qquad \text{B.6}$$

B.7.3.3 Normalized Beam Weights

Normalized beam weights are computed so their root-mean-square (RMS) sum is unity and quantized to realizable antenna settings (61 gain and phase states). The equation depicts their real (γ) and imaginary (δ) parts. The normalized beam weight vector [X] has the form:

Ith Element of Normalized Beam Weights {1,1}

$$X_i = \text{Quantize}\langle BW_i \,/\, \{\textstyle\sum_{61} |BW_i|^2\}^{1/2} \rangle \qquad \text{B.7}$$

Vector of Normalized Beam Weights {61,1}

$$X = \begin{vmatrix} \gamma_1 + j\delta_1 \\ \gamma_2 + j\delta_2 \\ \cdots \\ \cdots \\ \gamma_{61} + j\delta_{61} \end{vmatrix} \qquad \text{B.8}$$

AD-A163 530 DSCS NETWORK PERFORMANCE SOFTWARE SUPPORT REVISION(U) 2/2
M/A-COM LINKABIT INC VIENNA VA S V TREES ET AL.
07 JAN 86 DCA100-84-C-0009

UNCLASSIFIED F/G 9/2 NL

END
FILMED
DTIC

B.7.3.4 Computed Gain

The computed gain (or sensitivity) vector (G) is computed as the product of the singlet data matrix times the normalized beam weights vector:

<u>Computed Gain {M,1}</u>

$$G = A X \qquad \text{B.9}$$

B.7.3.5 Directive Gain

The resultant gain vector G is modified by the MBR_AAS to produce the contour data (directive gain (λ)). Directive gain is a direct function of the computed beam weights. MBR_AAS updates the beam weight vector X to make the computed gain vector lambda (λ) constrained by the desired gain bounds (Eq. B.4 and B.5). This equation scales directive gain into decibel units. Note that the gain G is defined in complex space whereas the directive gain vector (λ) is magnitude only.

<u>Directive Gain</u> {M,1}

$$\lambda_i = 10*\log_{10}(Sf*GG^*)_{i,i} \qquad \text{B.10}$$

where:

Sf	Scale Factor: GE Supplied
G	Equation B.9
G^*	The complex conjugate transpose of Equation B.9
i,i	Form a vector from the diagonal elements of the matrix
i	An element from the set of elements i = 1 to M

Expanding equations B.9 and B.10 into their composite elements is presented for clearity:

$$G_i = \Sigma_{k=1,61}\ (\alpha_{i,k}\gamma_k) - \beta_{i,k}\delta_k) + j(\beta_{i,k}\gamma_k + \alpha_{i,k}\delta_k) \qquad \text{B.10a}$$

$$G_i = Re(G_i) + jIm(G_i) \qquad \text{B.10b}$$

$$(G_i G_i^*)_{i,i} = (Re(G_i))^2 + (Im(G_i)^2) \qquad \text{B.10c}$$

where:

Re()	The real part of the complex vector
Im()	The imaginary part of the complex vector
j	The imaginary unit vector

B.7.3.6 First Partial Derivatives

Once the initial choice of the beam weight vector **X** is chosen, the first partial derivative of lambda (λ) is evaluated with respect to the magnitude of each directive gain element. Two sets of partial derivatives are evaluated for lambda with respect to the real (δ) and imaginary (γ) parts of the beam weight vector **X**. The partial derivatives are used by the steering algorithm to determine if the set of beam weights will add to, subtract from, or leave unchanged the directive gain vector λ for the areas under consideration. The first partial derivatives are computed by the matrix operation:

<u>Partial Derivatives</u> {M,61}

$$PD_{M,61} = \partial\lambda_M / \partial X_{61} \qquad \text{B.11}$$

In an expanded form the partial deratives are:

$$PD_\gamma = \sum_M \sum_{61} 2[-\beta^T Re(\lambda)+\alpha^T Im(\lambda)] \qquad \text{B.12}$$

$$PD_\delta = \sum_M \sum_{61} 2[\alpha^T Re(\lambda)+\beta^T Im(\lambda)] \qquad \text{B.13}$$

Where:

α,β	The real and imaginary elements of Equation B.3
α^T,β^T	The transpose of α,β
γ	The real part of the partial derative
δ	The imaginary part of the partial derivative
λ	Directive gain (Equation. B.10)
Re(),Im()	Functions that return the real and imaginary parts of the argument

B.7.4 Beam Selection Algorithm

Two measures are computed to determine if the beam weight vector selection **X** is valid: the constraint equation (Eq. B.15) and the beam quality index (Eq. B.16), which are both discussed below. The **X** vector is chosen in a straightforward manner. The choice is simply the best fit for one of the points (1 through **M**) irrespective of all other points; therefore, a solution is localized to that individual point from an area of interest.

The current MBR_AAS computes beam weight updates based on the beam quality equation (Eq. B.19). The magnitude of the first partial derivatives allows the beam quality algorithm to determine if the computed beam weight vector (**X**) biases the overall contour. The magnitude of the first partial derivative is a complex conjugate product:

$$|\, \partial\lambda/\partial X \,| = \left([Re(PD) + jIm(PD)]\ [Re(PD) - jIm(PD)] \right)^{1/2} \qquad B.14$$

where:

$\| \ldots \|$	Denotes a vector magnitude operator
$Re(\ldots)$	The real part of the vector
$Im(\ldots)$	The imaginary part of the vector

These magnitudes are used in two different criteria for the update, as is discussed in the beam quality section.

B.7.4.1 Constraint Equation {Scalar}

The constraint equation is a measure of how well the computed beam weights compare to the *realizable values* (*i.e., how well the updates conform to Equation B.8*). The constraint equation is not bounded by satellite-imposed constraints of discrete gain and phase states. Constraint beam weights are the algorithm computed beam weights. Equation B.8 accounts for satellite-imposed constraints. While the constraint equation is not used in beam weight updates, it supplies a measure between the computed beam weights and the realizable values. Equation B.15 scales beam weights to ±1000 for integer mathematics. If the computed beam weights ideally equal the normalized beam weights (Eq. B.8), then the sum of the magnitude squared would be one million. This may be defined as:

$$Tconst = \Sigma_{61}\ |\, XC + \Delta XC \,|^2 => 10^6 \text{ ; Exact Fit of Eq. B.8} \qquad B.15$$

where:

$\Sigma_{61} \| \ldots \|^2$	The sum of the vector magnitude squared
XC	The computed beam weights (BW times 1000)
ΔXC	The computed beam weight updates (BW times 1000)

B.7.4.2 Beam Quality Equation

Beam quality is the controlling variable in the beam weight update matrix. Two modes of control are used for variable[3] and constant update step size. A definition of beam quality is the measure of how well a particular set of beam weights biases the

---

[3]*The variable step size is a LINKABIT change.*

overall directive gain with respect to the desired results. Biasing in the MBR_AAS software is defined as changing the directive gain in a perverse direction by the overall contour being increased or decreased more than some desired amount. This amount is compared to epsilon (ε). Epsilon is a qualitative measure that sets an upper limit on the sum of perverse beam quality elements or "moves." If the sum of the beam quality elements (moves: +1, −1, or 0) is smaller than epsilon, the beam weight is accepted. If the sum is greater than epsilon, the beam weights are rejected and the MBR_AAS selects a new area of interest or enters an impasse upon beam weight rejection. Such biasing would tend to move the contour toward an unwanted result; hence, it is a rejection criterion. Biasing is compared to epsilon by:

<u>Beam Quality Vector {M,61}</u>

$$BC_M = \sum_M \quad DZ\{\lambda_M - Bnd\} \qquad \text{B.16}$$

where:

DZ{...}	Dead Zone sign of function: ±1, or 0
M	The area of interest
Bnd	Area definition bound limits (hi & low)
Bnd	= Boundlow when low bound not satisfied (Equation B.5)
Bnd	= Boundhi when low bound satisfied (Equation B.4)

$BC_M \leq \varepsilon$ - Accept beam weights

$BC_M > \varepsilon$ - Reject beam weights

Beam qualities are then sorted according to magnitude. The largest magnitude beam quality is used as the next area of interest. Updates to the beam weight vector are formed by the beam quality for the point of interest. The first mode of steering allows incremental beamweight updates and the second mode integrates the unity step size update. Equation B.18 was implemented to allow a proportional amount of beam quality to update the beam weights. The unity step size update (Equation B.17) was removed from MBR_AAS because none of the test cases converged with this choice. The following equations depict the two modes:

Incremental or Unity Step Update

$$XC_{k+1} = XC_k + BC_k \qquad \text{B.17}$$

Integrated Stepsize Update

$$XC_{k+1} = XC_k + \Sigma_{updates}\, BC_k \qquad \text{B.18}$$

where:

XC	Computed beam weights at iteration step k
BC	Computed beam quality at iteration step k
$\Sigma_{updates}$	Sum over all iteration steps (1 through k)

The dead zone sign function is shown in Figure B-4. It returns a value of zero or ±1. In the constant update beam quality criterion, the update is simply this value. In the variable update criterion, this value is integrated into (added to) the current update index.

Figure B-4. Dead Zone Sign Function

```
                          |
                          | DZ(lambda)
                          |1          +-------------
                          |           |  boundhi
                          |           |
-----------------------------------------/ lambda /------
           |              |
           |              |
---------+ boundlow       |-1
                          |
```

B.7.5 The Iteration Cycle of MBR_AAS

Equation B.18 represents beam weight updates as a function of beam quality. This equation needs to be normalized and quantized to implement the satellite-imposed constraints. The iteration loop is completed when Equation B.18 is normalized according to Eq. B.7 and B.8. The iteration continues until all the points within the areas of interest are constrained by their desired gains or the operator aborts the process.

B.7.6 Area Selection Algorithm

The area selection algorithm gives equal priority to all areas according to their size. This algorithm sorts areas (points of interest) in a unusual fashion. Points are defined within the areas of interest. They represent the singlet data encompassed by these areas. These points are labelled from 1 through M. The key to the algorithm is how they are selected. This is done by TEKSORT. TEKSORT sorts the points (M) by selecting them from the centroid of each area. It alternates the areas to choose from by their size. If, for instance, two areas are under consideration and each area contains the same number of points, it will alternate points from the centroid of each area. Each new selection will be the next closest point to the centroid of each area until all the points are evaluated. If the areas of interest are unequal in size, a proportional number of points will be evaluated according to size. An area twice a large as another area will have points evaluated twice as often as the smaller area. The proportionality rule holds true for all defined areas. This selection process queues the areas into the beam quality steering algorithm. When the current set of areas has been constrained to their desired gains, more areas are introduced from the area priority algorithm.

B.7.7 Case Studies of MBR_AAS

All case studies were derived by simple combinations of singlet beams. The beam weights were generated and verified by the DNPS scenario definition and contour map reporting functions. MBR feed elements used to generate the case studies are identified directly below the figure titles. This technique was used to generate realizable area/gain definitions for MBR_AAS input. Four case studies are presented which represent a combination of area/gain definitions inside and outside a ±4° look angle (or pointing angle). This angle is referenced from the satellite to the Earth location in terms of azimuth/elevation coordinates. The ±4° look angle was chosen to demonstrate MBR_AAS convergence properties. Figures B-5 through B-8 depict the area/gain definitions. A three-character alphanumeric descriptor is the label for area of interest (labelled "area"). The directive gain boundaries for the area of interest are labelled "above" for the high limit and "below" for the low limit. The cases with figure numbers are summarized as follows:

Figure Number	Gain (dB) Above	Gain (dB) Below	Area Description
			Case I
B-5	23	27	"C1A" terminal inside ±4° look angle
	17	13	"C1D" terminal cut by ±4° look angle
			Case II
B-6	27	23	"C2A" terminal cut by ±4° look angle
	12	8	"C2B" terminal inside ±4° look angle
	27	23	"C2C" terminal outside ±4° look angle
	17	13	"C2D" terminal outside ±4° look angle
			Case III
B-7	22	18	"C3A" terminal inside ±4° look angle
	7	3	"C3B" jammer inside ±4° look angle
			Case IV
B-8	27	23	"C4A" terminal outside ±4° look angle

Figure B-5. Case I Terminal Locations Aligned with Singlet Beams Singlet Beams 16, 40, 41, & 48

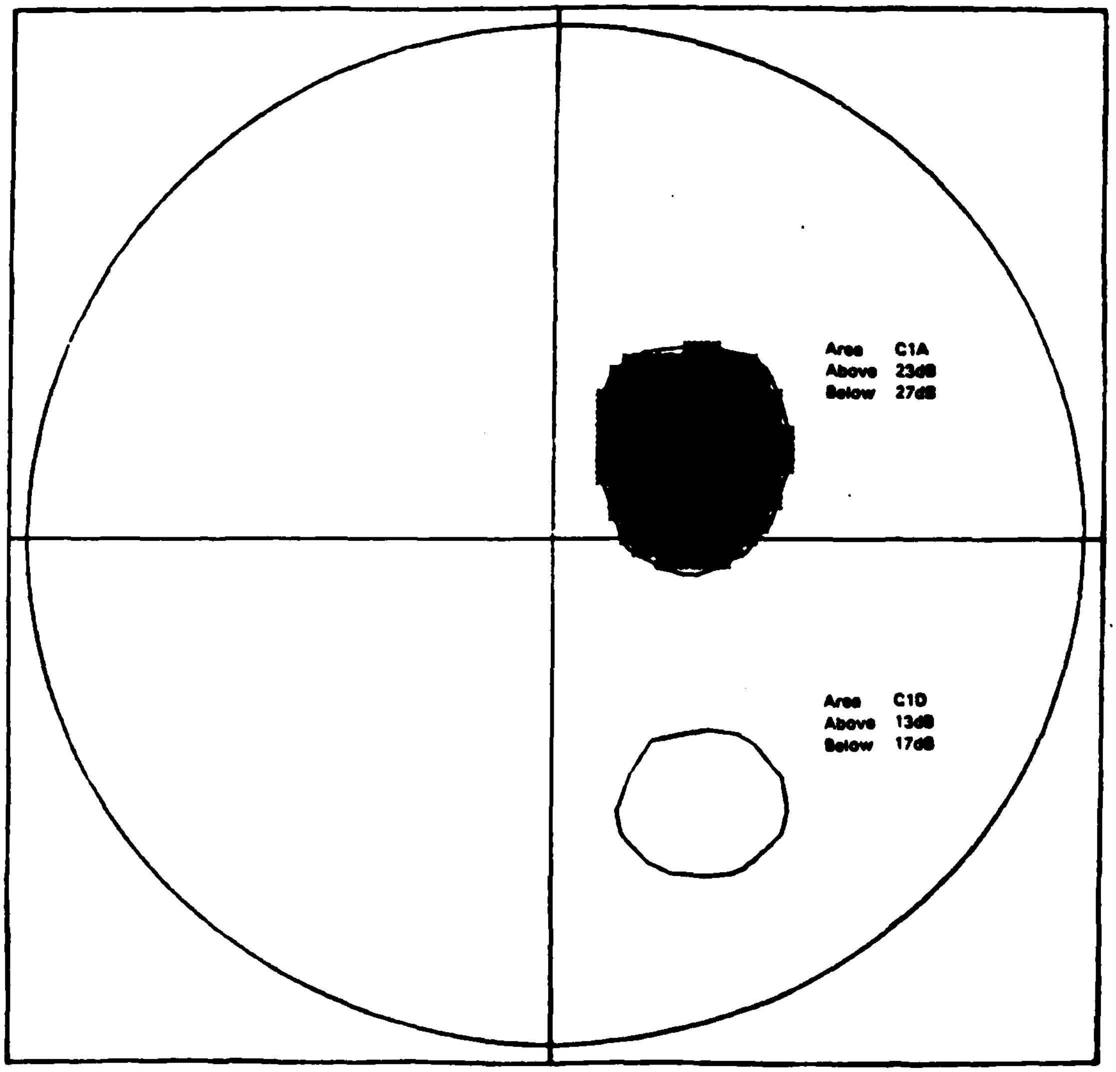


Figure B-6. Case II Dispersed Terminal Locations Singlet Beams 10, 13, & 22

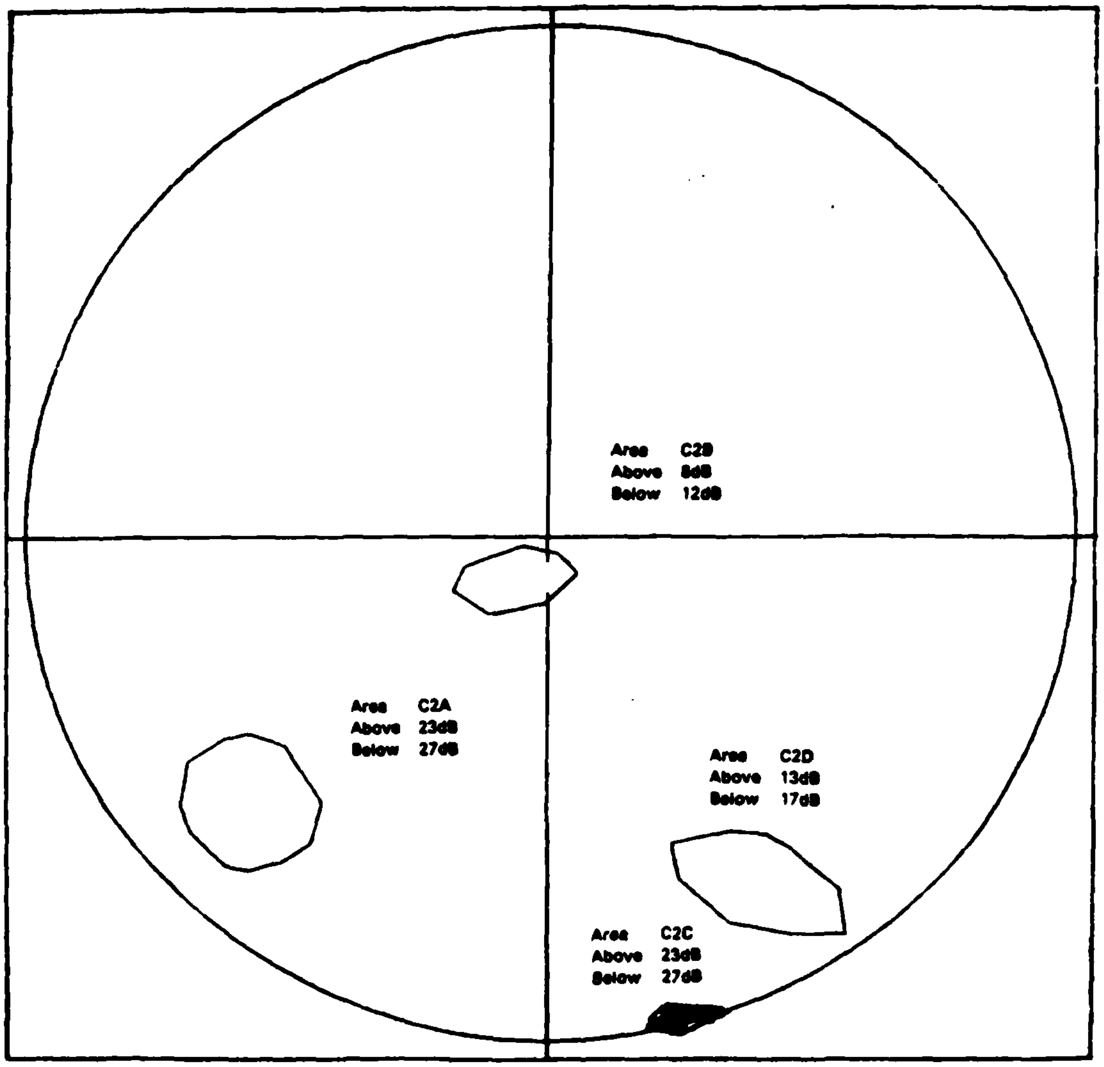


Figure B-7. Case III Terminal/Jammer Inside Converging Look Angle Singlet Beams 16, 22, 23, & 32

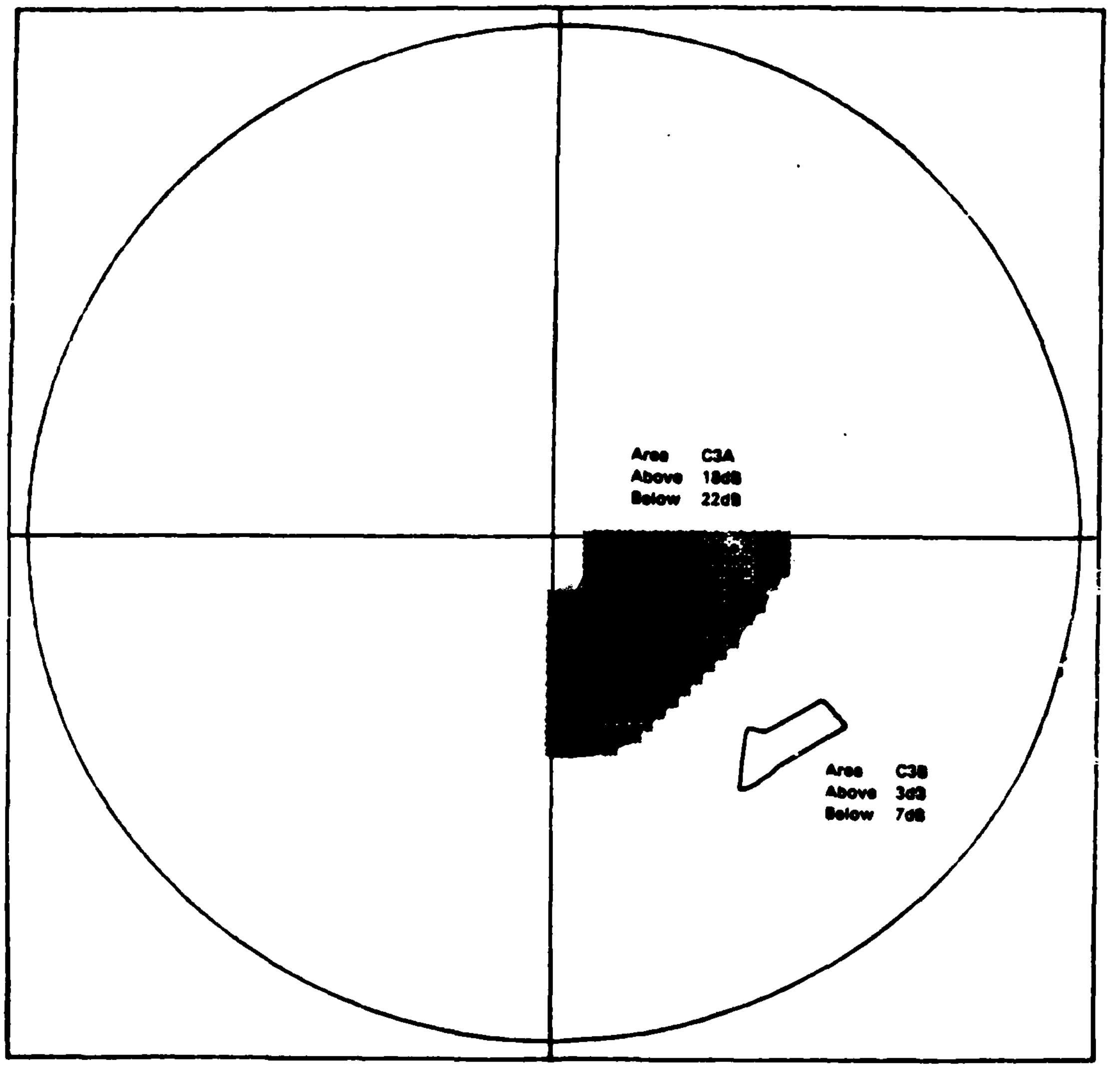
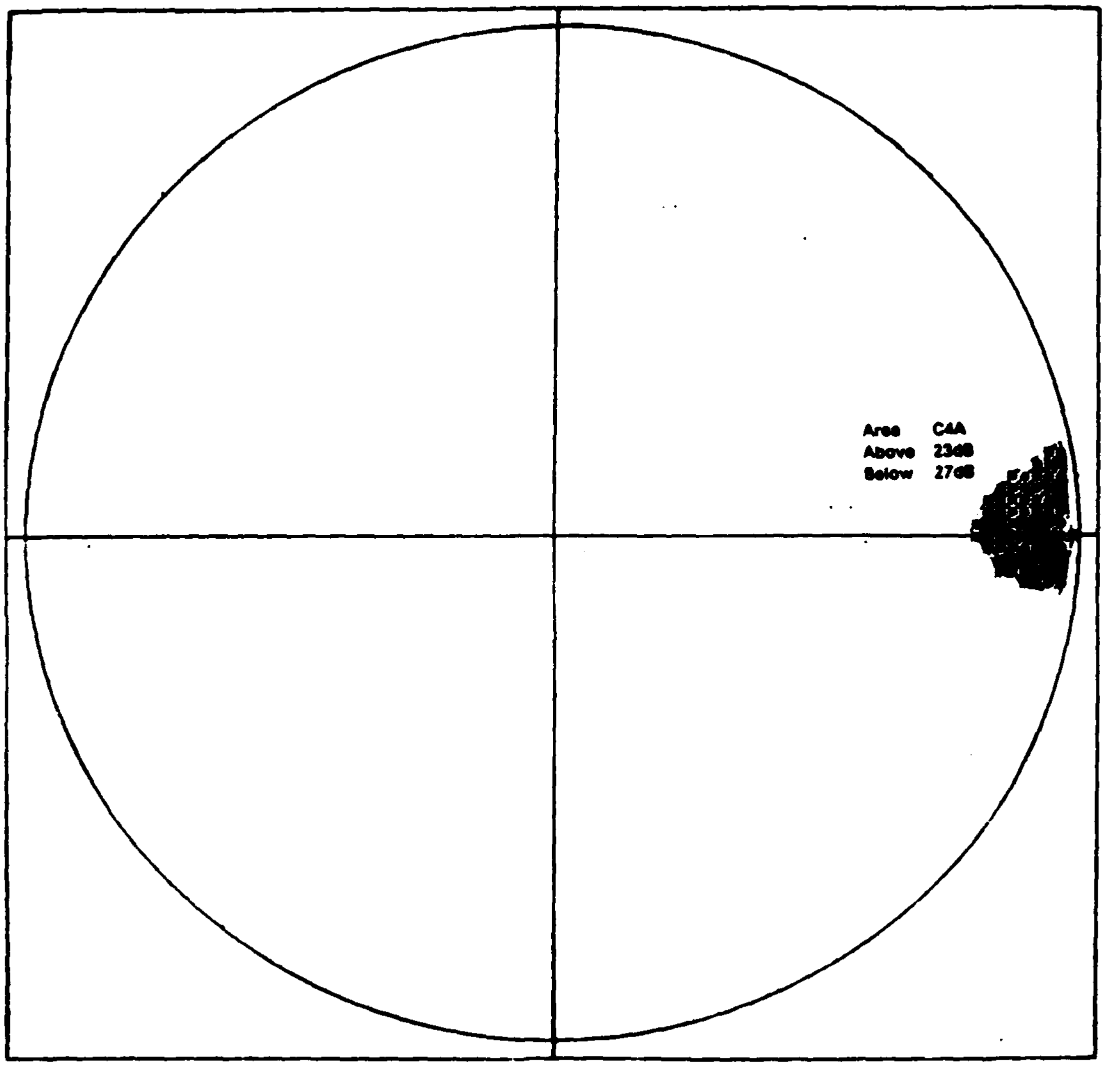


Figure B-8. Case IV Terminal Outside Converging Look Angle Singlet Beam 34



Case I was designed to test the correlation of singlet beam boresights with areas of interest. Area "C1A" is centered in between the boresights of singlet beams 40, 41, and 48. Area "C2A" is centered under the boresight of singlet beam 16. Desired gains were chosen so the four beam weights are equal magnitude with zero phase. Over half of the 61 beams contributed dominant magnitude and phase components to the solution. This confirms the lack of algorithm sensitivity toward singlet beam illumination directly under the areas of interest.

Case II was designed to test a combination of converging and nonconverging area/gain definitions, and the area priority sorting algorithm (TEKSORT). This case has areas defined inside, outside, and cut by the ±4° look angle criterion. The algorithm failed to converge after 1 hour of execution time. Not one point from area "C2C" was considered in this time. This test demonstrates that the algorithm cannot combine converging and nonconverging areas of interest.

Case III was designed as a terminal/jammer scenario. The terminal and jammer locations were widely separated to avoid any tight constraint Both areas were placed inside the ±4° look angle to avoid convergence problems. This case converged after 2 hours and 20 minutes of computation time.

Case IV was designed to test areas defined under a large look angle. Area "C4A" is defined just above the horizon. After 1 hour of computation time, no points met the constraints.

B.7.8 Conclusions on MBR_AAS Performance

The test cases show that MBR_AAS performs marginally for simple area/gain definitions. A general observation from Table B.1 shows that high gain areas develop outside the defined areas. Such undesired gain demonstrates lack of overall power management within the algorithm. In summary, four distinct problems are:

- No convergence outside a ±4° look angle
- No main lobe preference for areas directly under a singlet beam
- Undesired power allocation

Table B-1. Summary of Beamweights for Cases I through IV

Beam	This is Case I Beamweights		This is Case "II" Beam Weights		This is Case III Beam Weights		This is case IV Beam Weights	
beam(1)	0.148	0.016	0.029	0.063	0.165	0.066	0.018	0.115
beam(2)	0.017	0.017	0.182	0.065	0.056	0.085	0.168	0.141
beam(3)	0.050	0.148	0.027	0.029	0.134	0.123	0.018	0.020
beam(4)	0.016	0.175	0.156	0.153	0.060	0.031	0.225	0.141
beam(5)	0.016	0.122	0.275	0.027	0.113	0.063	0.030	0.225
beam(6)	0.016	0.175	0.194	0.192	0.063	0.050	0.225	0.168
beam(7)	0.046	0.148	0.030	0.065	0.031	0.050	0.141	0.018
beam(8)	0.175	0.175	0.343	0.027	0.096	0.063	0.029	0.039
beam(9)	0.147	0.037	0.030	0.065	0.050	0.129	0.141	0.168
beam(10)	0.175	0.016	0.221	0.252	0.165	0.063	0.016	0.020
beam(11)	0.148	0.148	0.061	0.065	0.050	0.077	0.018	0.164
beam(12)	0.147	0.148	0.028	0.372	0.112	0.128	0.018	0.225
beam(13)	0.148	0.016	0.182	0.114	0.063	0.165	0.115	0.018
beam(14)	0.017	0.017	0.038	0.067	0.123	0.056	0.018	0.167
beam(15)	0.148	0.148	0.525	0.016	0.034	0.161	0.225	0.018
beam(16)	0.045	0.148	0.365	0.041	0.050	0.032	0.018	0.032
beam(17)	0.015	0.175	0.016	0.029	0.063	0.069	0.225	0.225
beam(18)	0.028	0.147	0.030	0.098	0.077	0.050	0.018	0.020
beam(19)	0.148	0.028	0.065	0.171	0.034	0.142	0.225	0.225
beam(20)	0.017	0.017	0.041	0.306	0.063	0.112	0.168	0.225
beam(21)	0.148	0.148	0.142	0.182	0.128	0.031	0.018	0.020
beam(22)	0.148	0.175	0.063	0.068	0.032	0.032	0.033	0.225
beam(23)	0.017	0.016	0.029	0.182	0.165	0.142	0.225	0.225
beam(24)	0.017	0.017	0.039	0.021	0.056	0.069	0.072	0.018
beam(25)	0.148	0.066	0.252	0.182	0.063	0.031	0.018	0.225
beam(26)	0.175	0.028	0.525	0.182	0.050	0.315	0.018	0.115
beam(27)	0.016	0.017	0.029	0.039	0.165	0.244	0.018	0.031
beam(28)	0.175	0.029	0.252	0.122	0.165	0.056	0.018	0.141
beam(29)	0.122	0.175	0.029	0.029	0.035	0.069	0.225	0.115
beam(30)	0.147	0.017	0.175	0.127	0.076	0.161	0.225	0.168
beam(31)	0.148	0.175	0.027	0.028	0.063	0.161	0.168	0.018
beam(32)	0.148	0.175	0.053	0.252	0.128	0.123	0.225	0.020
beam(33)	0.016	0.016	0.306	0.029	0.095	0.035	0.018	0.141
beam(34)	0.175	0.147	0.115	0.127	0.233	0.023	0.225	0.225
beam(35)	0.175	0.148	0.463	0.252	0.063	0.076	0.018	0.225
beam(36)	0.148	0.057	0.053	0.252	0.050	0.032	0.033	0.225
beam(37)	0.029	0.016	0.028	0.182	0.022	0.045	0.141	0.033
beam(38)	0.017	0.148	0.028	0.047	0.112	0.056	0.225	0.031
beam(39)	0.017	0.148	0.252	0.182	0.165	0.082	0.031	0.225
beam(40)	0.175	0.148	0.029	0.525	0.034	0.031	0.225	0.225
beam(41)	0.175	0.175	0.028	0.030	0.112	0.112	0.031	0.018
beam(42)	0.016	0.016	0.016	0.069	0.056	0.112	0.225	0.225
beam(43)	0.016	0.148	0.029	0.050	0.165	0.069	0.225	0.115
beam(44)	0.148	0.015	0.182	0.029	0.127	0.142	0.031	0.141
beam(45)	0.175	0.044	0.029	0.041	0.144	0.032	0.225	0.225
beam(46)	0.148	0.016	0.030	0.252	0.032	0.161	0.020	0.056
beam(47)	0.016	0.147	0.039	0.029	0.063	0.161	0.196	0.018
beam(48)	0.147	0.175	0.029	0.041	0.056	0.069	0.016	0.225
beam(49)	0.015	0.028	0.030	0.041	0.050	0.032	0.225	0.225
beam(50)	0.017	0.147	0.064	0.030	0.068	0.035	0.018	0.019
beam(51)	0.148	0.175	0.053	0.049	0.050	0.069	0.018	0.225
beam(52)	0.015	0.016	0.182	0.064	0.123	0.069	0.225	0.225
beam(53)	0.057	0.148	0.252	0.028	0.126	0.031	0.018	0.018
beam(54)	0.147	0.148	0.029	0.209	0.288	0.160	0.018	0.225
beam(55)	0.147	0.122	0.033	0.041	0.123	0.142	0.018	0.018
beam(56)	0.016	0.016	0.029	0.182	0.112	0.056	0.017	0.225
beam(57)	0.122	0.175	0.333	0.182	0.165	0.031	0.018	0.225
beam(58)	0.175	0.017	0.525	0.065	0.035	0.161	0.225	0.018
beam(59)	0.016	0.175	0.281	0.171	0.041	0.112	0.019	0.018
beam(60)	0.016	0.122	0.252	0.030	0.032	0.101	0.225	0.225
beam(61)	0.175	0.122	0.016	0.030	0.038	0.066	0.225	0.018

• No full convergence for any definition ever achieved

B.7.9 Status of MBR_AAS

Two Multiple Beam Receive Resource Allocator Algorithm packages exist: DCEC's original model (RETMBR) and the LINKABIT working package (MBR_AAS). The original model did not converge for any of the four test cases; hence, no observations were presented. The LINKABIT version includes two changes. Area definition arrays have been increased from 300 to 4000 points. (A point is defined as one element of ground coverage defined as an azimuth/elevation coordinate.) This changed the possible areas of interest from roughly 6% to 75% of Earth coverage. Allowable step sizes for the beam weight matrix have been modified to change from a constant (0,±1) to an integrated step size as described by Eq. B.18. These changes enhance the program capacity and convergence rate but do not warrant serious consideration for the DNPS.

Appendix C

DETAILED EXPLANATION OF CASE STUDIES FOR MBR_AAS

This appendix extends the results of Section B.7 in Appendix B and provides a detailed explanation of the performance and convergence properties of the MBR_AAS software. The Figures B-3 through B-6 and Table B-1 are repeated for reader convenience as Figures C-1 through C-4 and Table C-1, respectively. Figures C-5 through C-13 represent intermediate results of the algorithms from Eq. B.7 and B.16. MBR_AAS creates these reports to display the convergence properties of the DCEC-provided algorithm at each iteration update as explained below.

Figure C-5 is a time snapshot of four Tektronix 4014 display updates as the MBR_AAS iterates through its first four iteration cycles. The leftmost number (4881) represents the current point under analysis. This is the record number for the azimuth/elevation number under consideration from the area labelled "C1A" (see Figure C-1). Record 4881 represents the point nearest centroid of area "C1A". Immediately following the area label (C1A) is a one-character summary field. It represents a summary of the beam quality equation (Equation B.16). This character may take on the following values:

I	The beam quality is below an acceptable value
A	The beam quality is above an acceptable value
J	The beam quality just recently became an acceptable value
B	The beam quality just recently became an unacceptable value
<u>Space</u>	The beam quality has been acceptable for several updates

The next set of 122 symbols consists of the elements of the dead zone sign function (Equation B.16) with 1 through 61 imaginary components and 1 through 61 real components. The initial row is all zeros, since no comparison can be made. Following rows have combinations of zero, for no impact on beam quality; a plus sign (+) for increased beam quality, a minus sign (-), for decreased beam quality; or a zero (0) for no change, or a space ( ), indicating that the beam quality has been acceptable for several iterations. The rows immediately above the beam quality are the computed beam weights (Equation. B.18) The 122 numbers represent the imaginary and real components (reading from the left to the right). For example, the first three imaginary beam weights for the third update are 6, 3, and 6. Figure C-6 through C-13 detail these types of data for the case studies. Typical figures display more areas of interest (up to 58) as the update

Figure C-1. Case I Terminal Locations Aligned with Singlet Beams

Singlet Beams 16, 40, 41, & 48

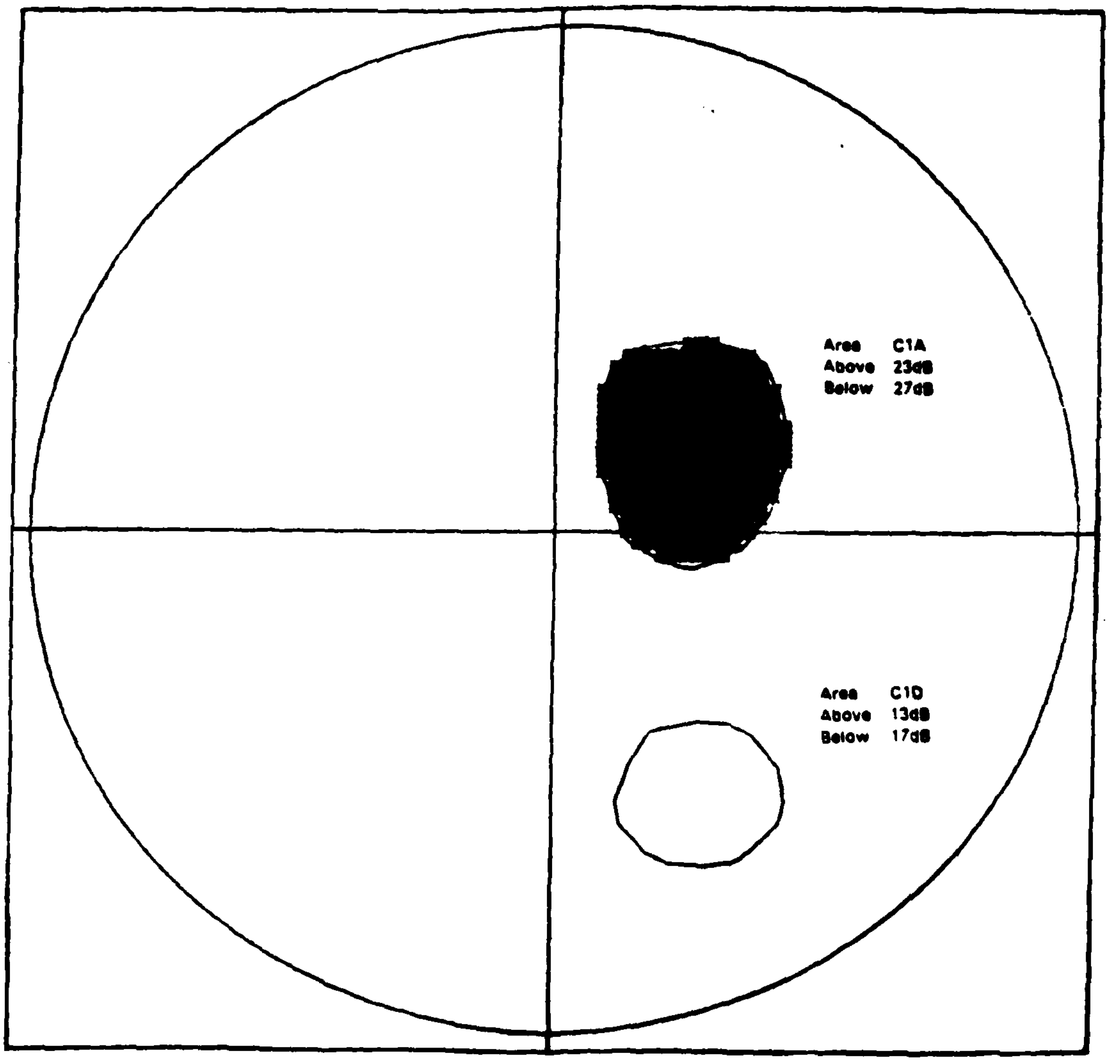


Figure C-2. Case II Dispersed Terminal Locations

Singlet Beams 4, 13, & 22

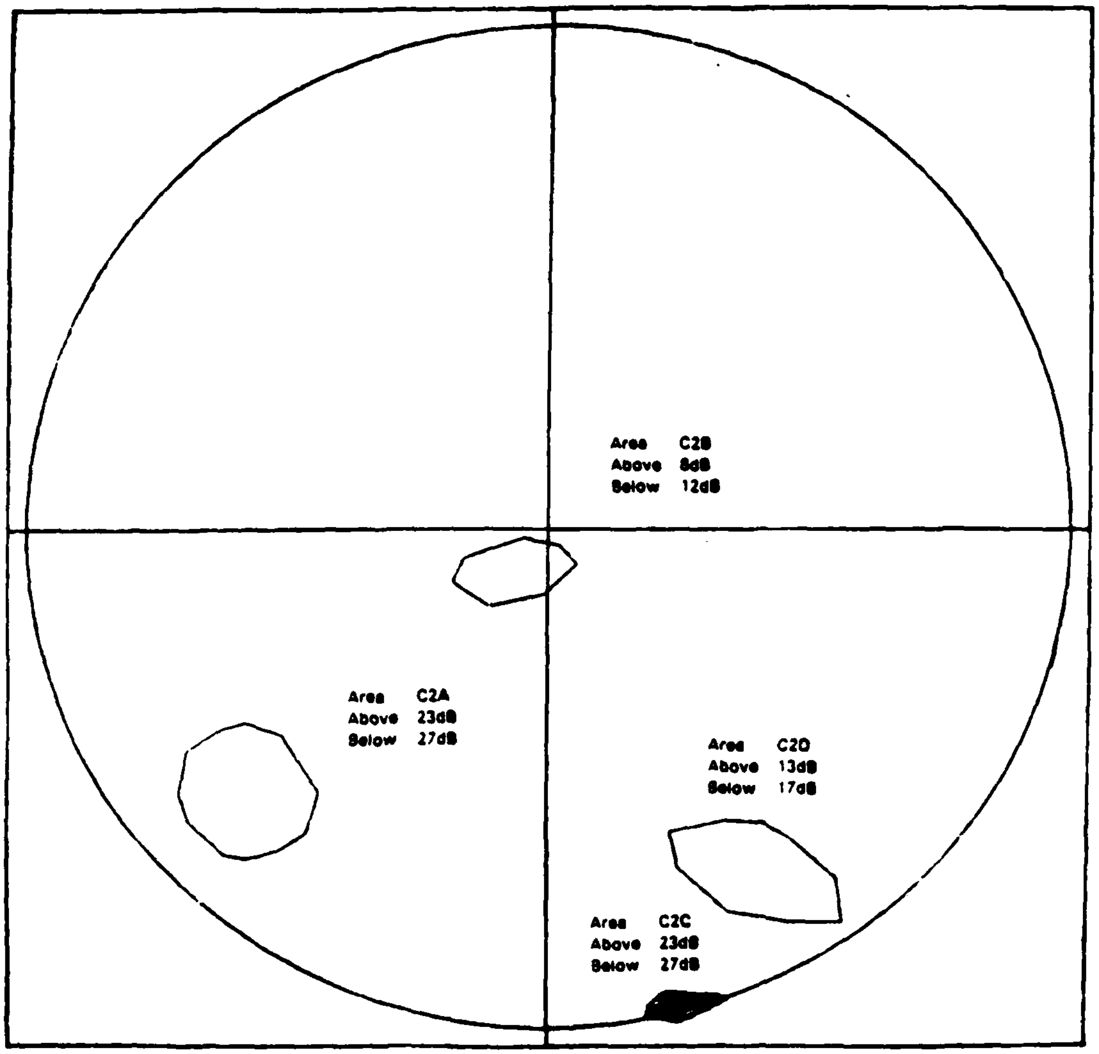


Figure C-3. Case III Terminal/Jammer Inside Converging Look Angle

Singlet Beams 16, 22, 23, & 32

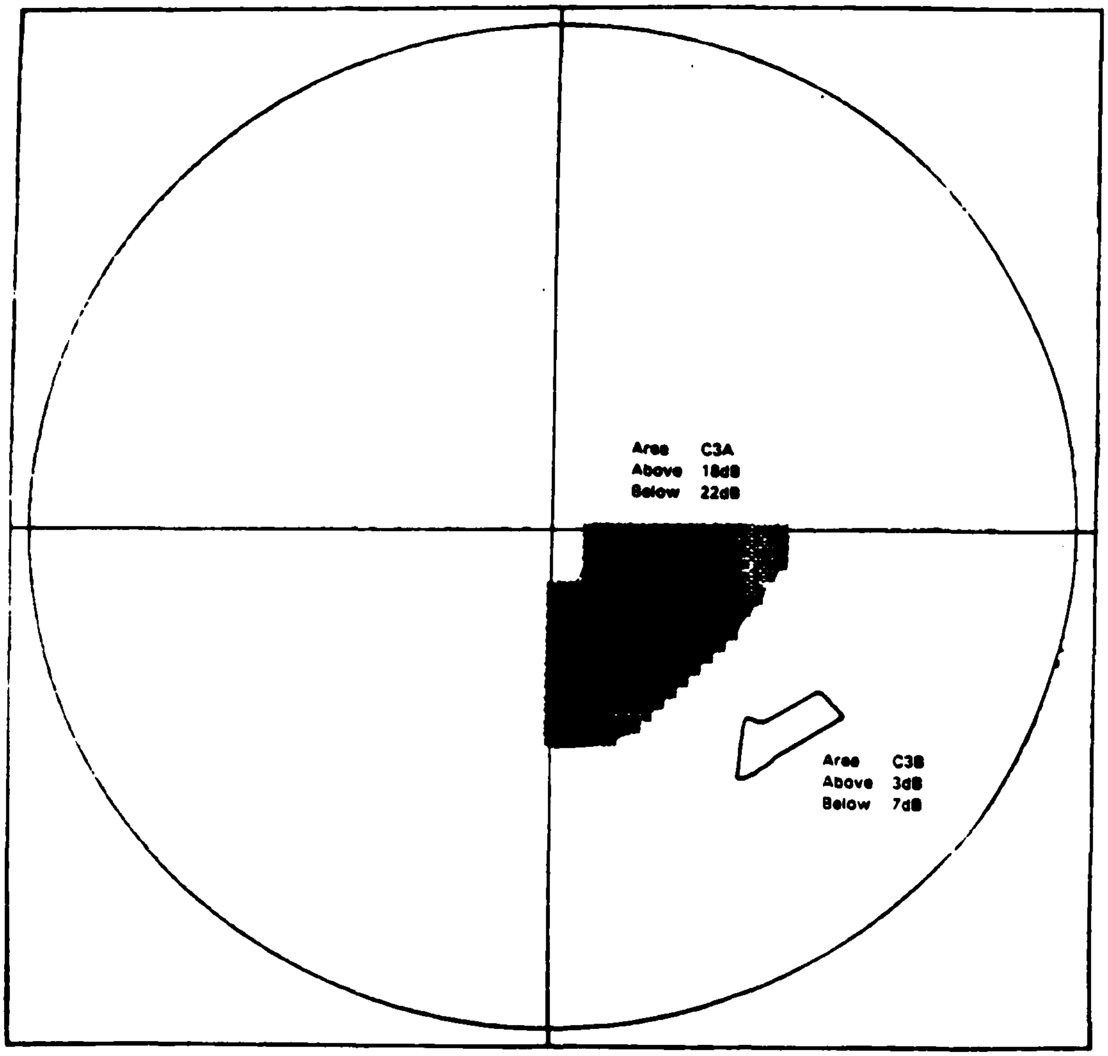


Figure C-4. Case IV Terminal Outside Converging look angle

Singlet Beam 34

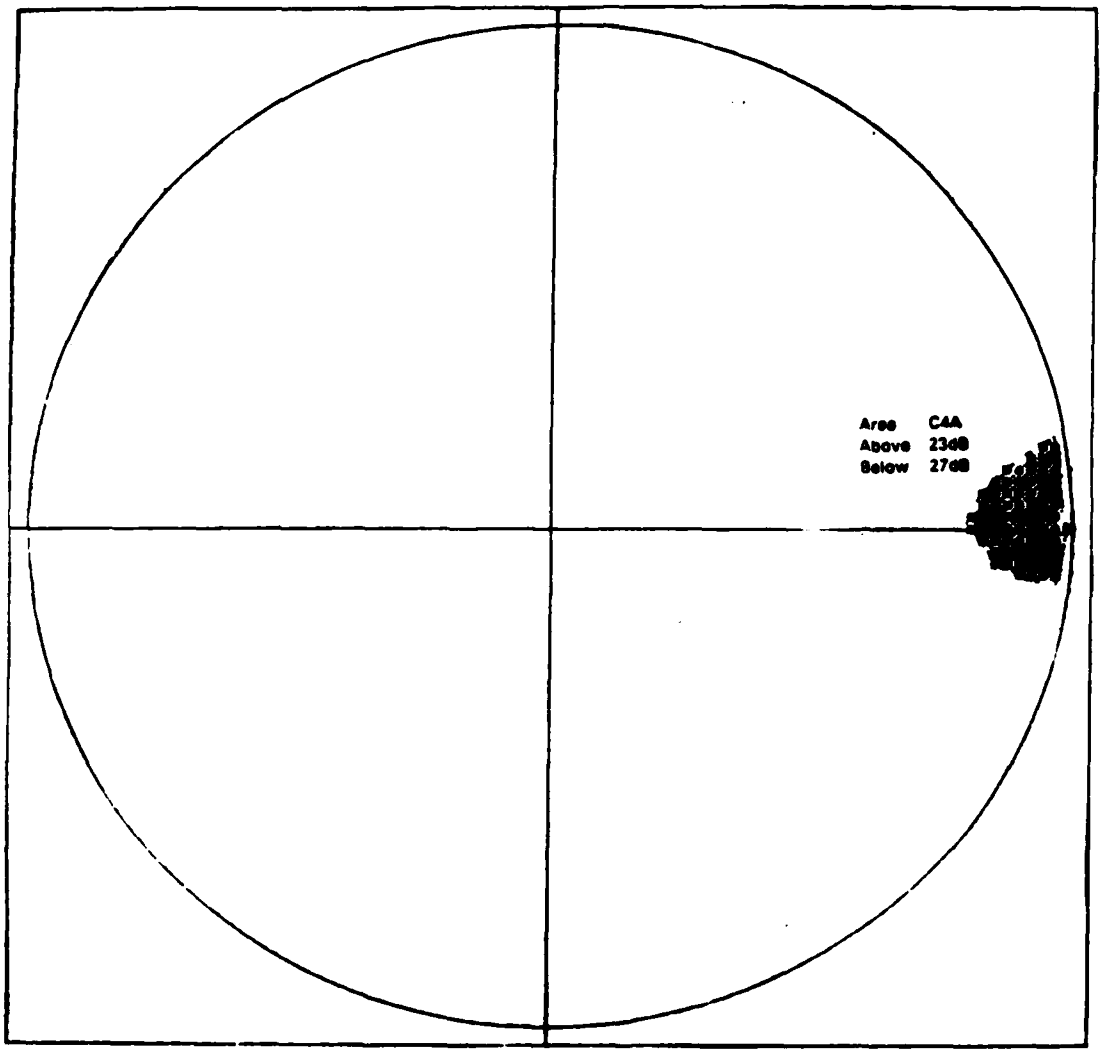


Table C-1. Summary of Beamweights for Cases I through IV

This is Case I Beamweights			This is Case ""II" Beam Weights		This is Case III Beam Weights		This is case IV Beam Weights	
beam(1)	0.148	0.016	0.029	0.053	0.165	[illegible]	0.018	0.115
beam(2)	0.017	0.017	0.182	0.065	[illegible]	[illegible]	[illegible]	0.141
beam(3)	0.060	0.148	0.027	0.029	0.134	0.123	0.018	[illegible]
beam(4)	0.016	0.175	0.156	0.153	[illegible]	0.031	[illegible]	0.141
beam(5)	0.016	0.182	0.275	0.027	0.113	0.063	[illegible]	[illegible]
beam(6)	0.016	0.175	0.194	0.192	0.063	0.050	[illegible]	[illegible]
beam(7)	0.046	0.148	0.030	0.065	0.031	0.050	0.141	0.018
beam(8)	0.175	0.175	0.343	0.027	[illegible]	0.063	[illegible]	0.030
beam(9)	0.147	0.037	0.030	0.065	0.050	0.128	0.141	0.168
beam(10)	0.175	0.016	0.221	0.252	0.165	0.063	0.016	0.020
beam(11)	0.148	0.148	0.061	0.065	0.050	0.077	0.018	0.164
beam(12)	0.147	0.148	0.028	0.372	0.112	0.128	0.018	0.225
beam(13)	0.148	0.016	0.182	0.114	0.063	0.165	0.115	0.018
beam(14)	0.017	0.017	0.038	0.067	0.123	0.056	0.018	0.167
beam(15)	0.148	0.148	0.525	0.016	0.034	0.161	0.225	0.018
beam(16)	0.045	0.148	0.065	0.041	0.050	0.032	0.018	0.032
beam(17)	0.015	0.175	0.016	0.029	0.063	0.069	0.225	0.225
beam(18)	0.028	0.147	0.030	0.098	0.077	0.050	0.018	0.020
beam(19)	0.148	0.028	0.065	0.171	0.034	0.142	0.225	0.226
beam(20)	0.017	0.017	0.041	0.306	0.063	0.112	0.168	0.226
beam(21)	0.148	0.148	0.142	0.182	0.128	0.031	0.018	0.020
beam(22)	0.148	0.175	0.063	0.068	0.032	0.032	0.033	0.225
beam(23)	0.017	0.016	0.029	0.182	0.165	0.142	0.225	0.225
beam(24)	0.017	0.017	0.039	0.021	0.056	0.069	0.072	0.018
beam(25)	0.148	0.066	0.252	0.182	0.063	0.031	0.018	0.226
beam(26)	0.175	0.028	0.525	0.182	0.050	0.315	0.018	0.115
beam(27)	0.016	0.017	0.029	0.039	0.165	0.244	0.018	0.031
beam(28)	0.175	0.029	0.252	0.122	0.165	0.056	0.018	0.141
beam(29)	0.122	0.175	0.029	0.029	0.035	0.069	0.225	0.115
beam(30)	0.147	0.017	0.175	0.127	0.076	0.161	0.225	0.168
beam(31)	0.148	0.175	0.027	0.028	0.063	0.161	0.168	0.018
beam(32)	0.148	0.175	0.053	0.252	0.128	0.123	0.225	0.020
beam(33)	0.016	0.016	0.306	0.029	0.095	0.035	0.018	0.141
beam(34)	0.175	0.147	0.115	0.127	0.233	0.023	0.225	0.226
beam(35)	0.175	0.148	0.463	0.252	0.063	0.076	0.018	0.225
beam(36)	0.148	0.057	0.053	0.252	0.050	0.032	0.033	0.225
beam(37)	0.029	0.016	0.028	0.182	0.022	0.045	0.141	0.033
beam(38)	0.017	0.148	0.028	0.047	0.112	0.056	0.225	0.031
beam(39)	0.017	0.148	0.252	0.182	0.165	0.082	0.031	0.226
beam(40)	0.175	0.148	0.029	0.525	0.034	0.031	0.225	0.225
beam(41)	0.175	0.175	0.028	0.030	0.112	0.112	0.031	0.018
beam(42)	0.016	0.016	0.016	0.069	0.056	0.112	0.225	0.226
beam(43)	0.016	0.148	0.029	0.050	0.165	0.069	0.226	0.115
beam(44)	0.148	0.015	0.182	0.029	0.127	0.142	0.031	0.141
beam(45)	0.175	0.044	0.029	0.041	0.144	0.032	0.225	0.226
beam(46)	0.148	0.016	0.030	0.252	0.032	0.161	0.020	0.056
beam(47)	0.016	0.147	0.039	0.029	0.063	0.161	0.196	0.018
beam(48)	0.147	0.175	0.029	0.041	0.056	0.069	0.016	[illegible]
beam(49)	0.015	0.028	0.030	0.041	0.050	0.032	0.226	[illegible]
beam(50)	0.017	0.147	0.064	0.030	[illegible]	0.035	0.018	0.019
beam(51)	0.148	0.175	0.063	0.040	0.050	0.069	0.018	[illegible]
beam(52)	0.015	0.016	0.182	0.064	0.123	0.069	[illegible]	[illegible]
beam(53)	0.057	0.148	0.252	[illegible]	0.165	0.031	0.018	0.018
beam(54)	0.147	0.148	[illegible]	[illegible]	[illegible]	0.100	0.018	[illegible]
beam(55)	0.147	[illegible]	0.033	0.041	0.123	0.142	0.018	0.018
beam(56)	0.016	0.016	[illegible]	0.182	0.112	[illegible]	0.017	[illegible]
beam(57)	[illegible]	0.175	[illegible]	0.182	0.165	0.031	0.018	[illegible]
beam(58)	0.175	0.017	0.525	[illegible]	0.035	0.161	[illegible]	0.018
beam(59)	0.015	0.175	[illegible]	0.171	0.041	0.112	0.019	0.018
beam(60)	0.016	[illegible]	[illegible]	0.030	[illegible]	0.101	[illegible]	[illegible]
beam(61)	0.175	[illegible]	0.016	[illegible]	[illegible]	[illegible]	[illegible]	0.018

Figure C-5. Tektronix 4014 Display Output (Case I Example)

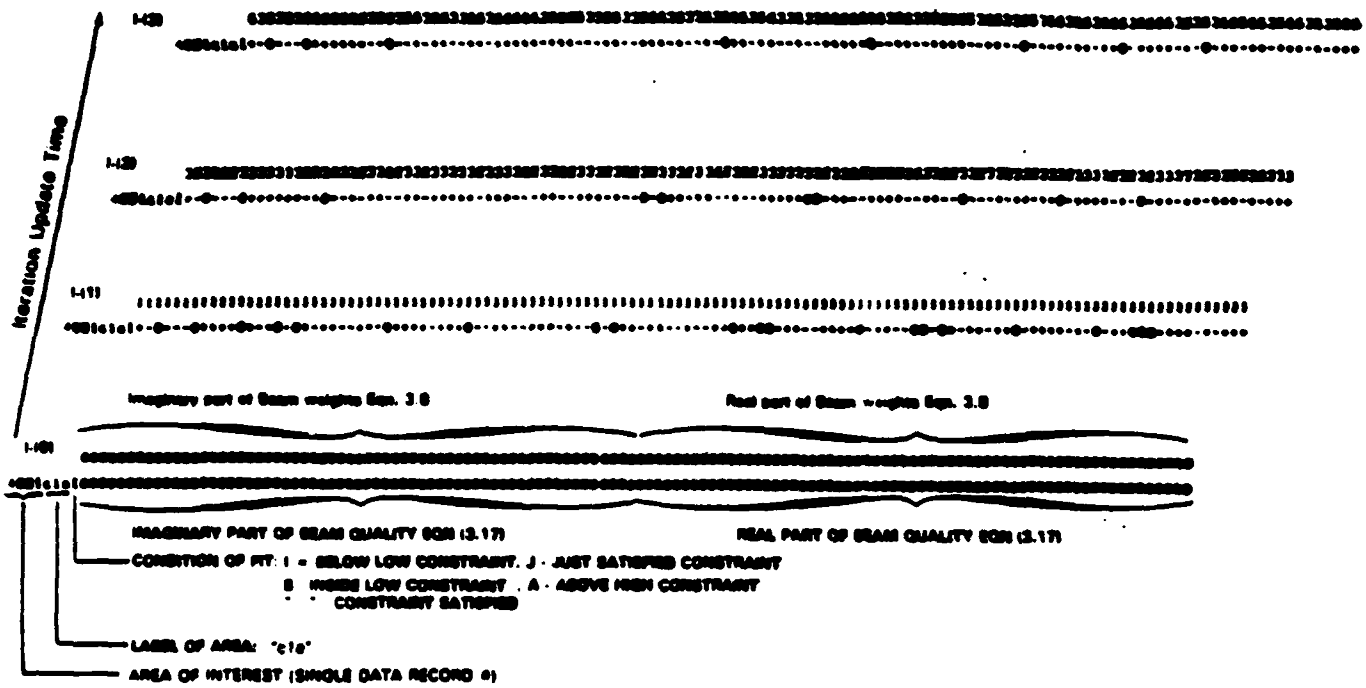


Figure C-6. Case I Fully Converged Constraints

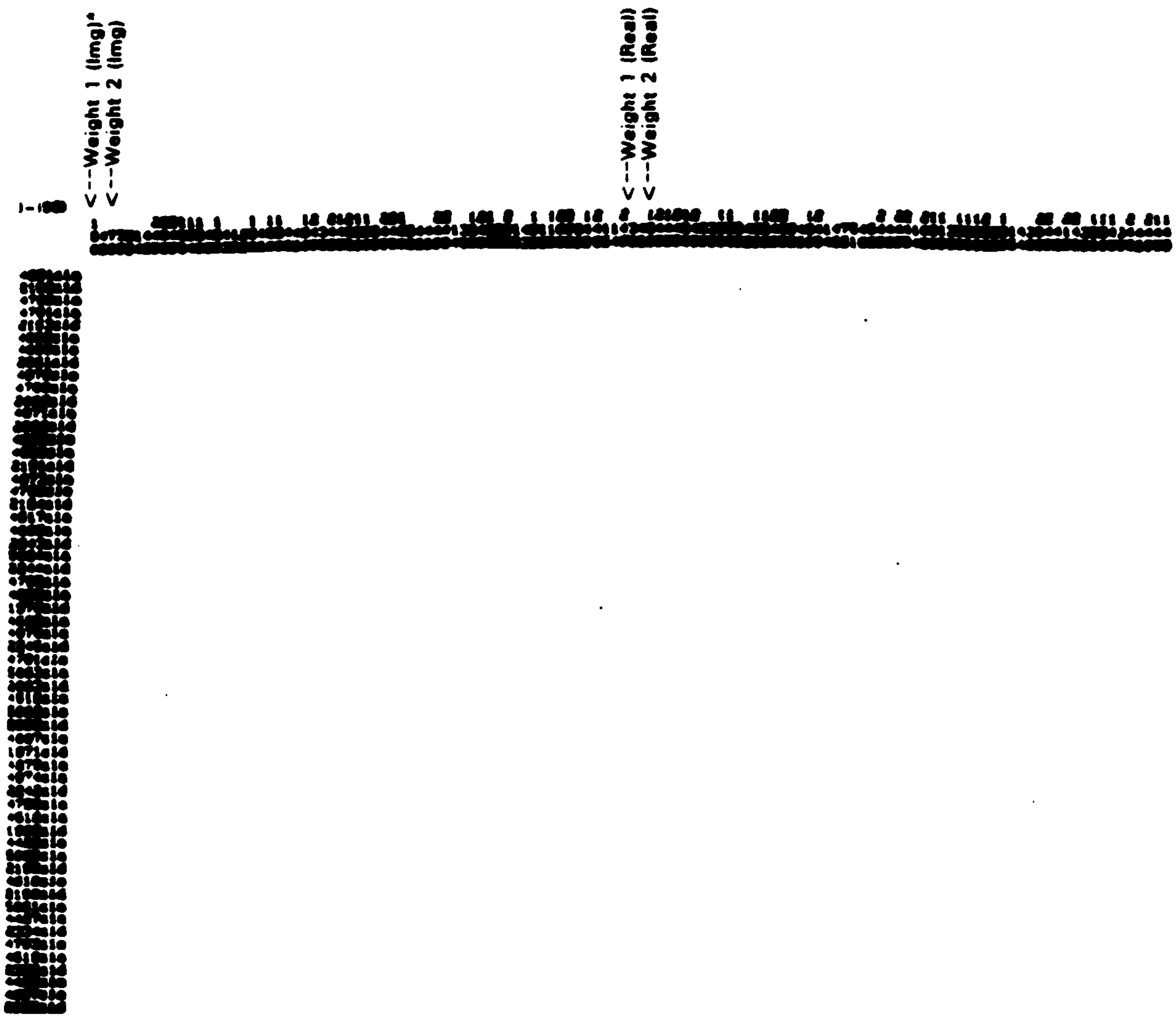
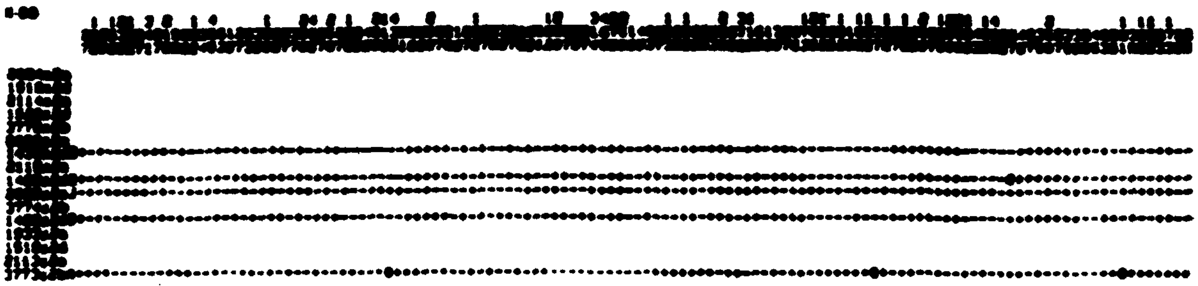


Figure C-7. Case II Updates 45 and 80

CASE II UPDATE 80



CASE II 45

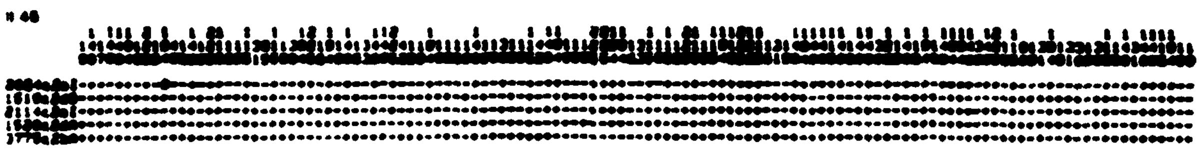


Figure C-8. Case II Updates 98 and 180

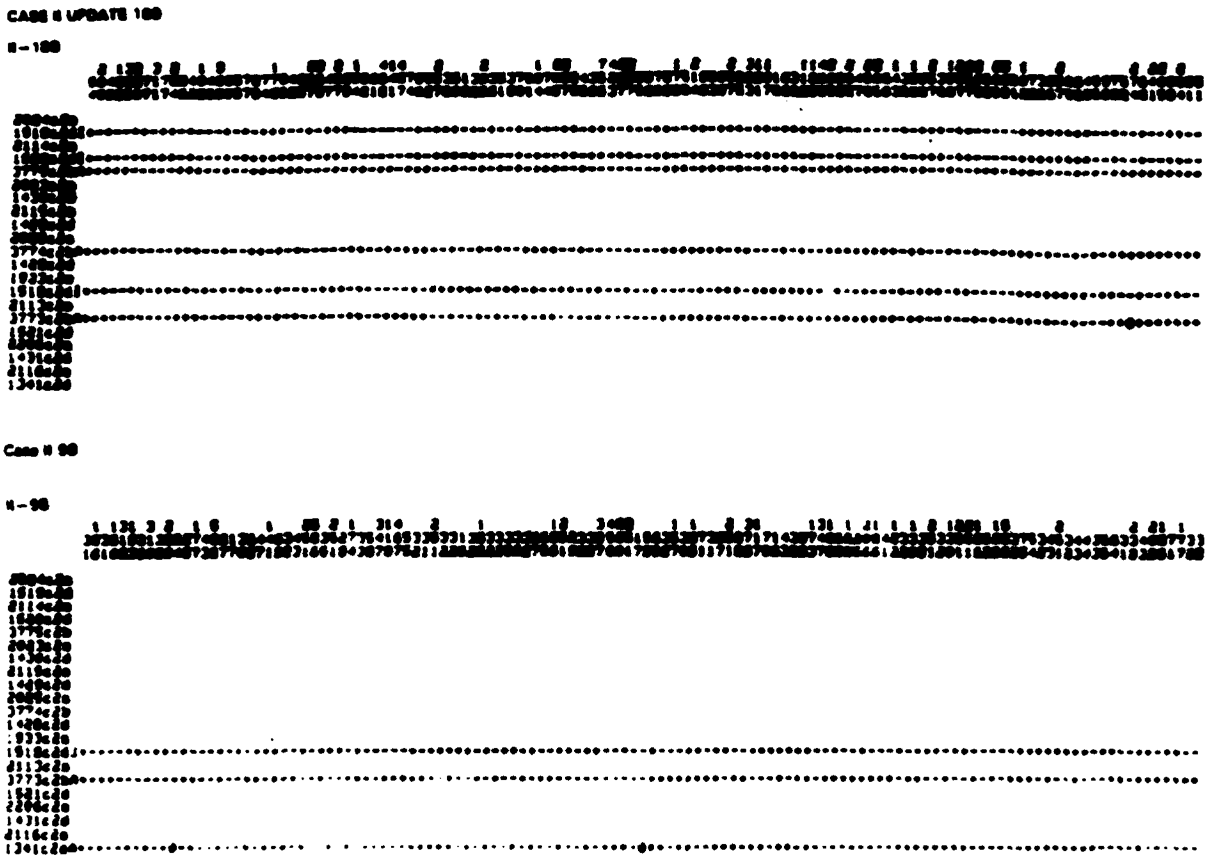


Figure C-9. Case III Updates 50, 113 and 160

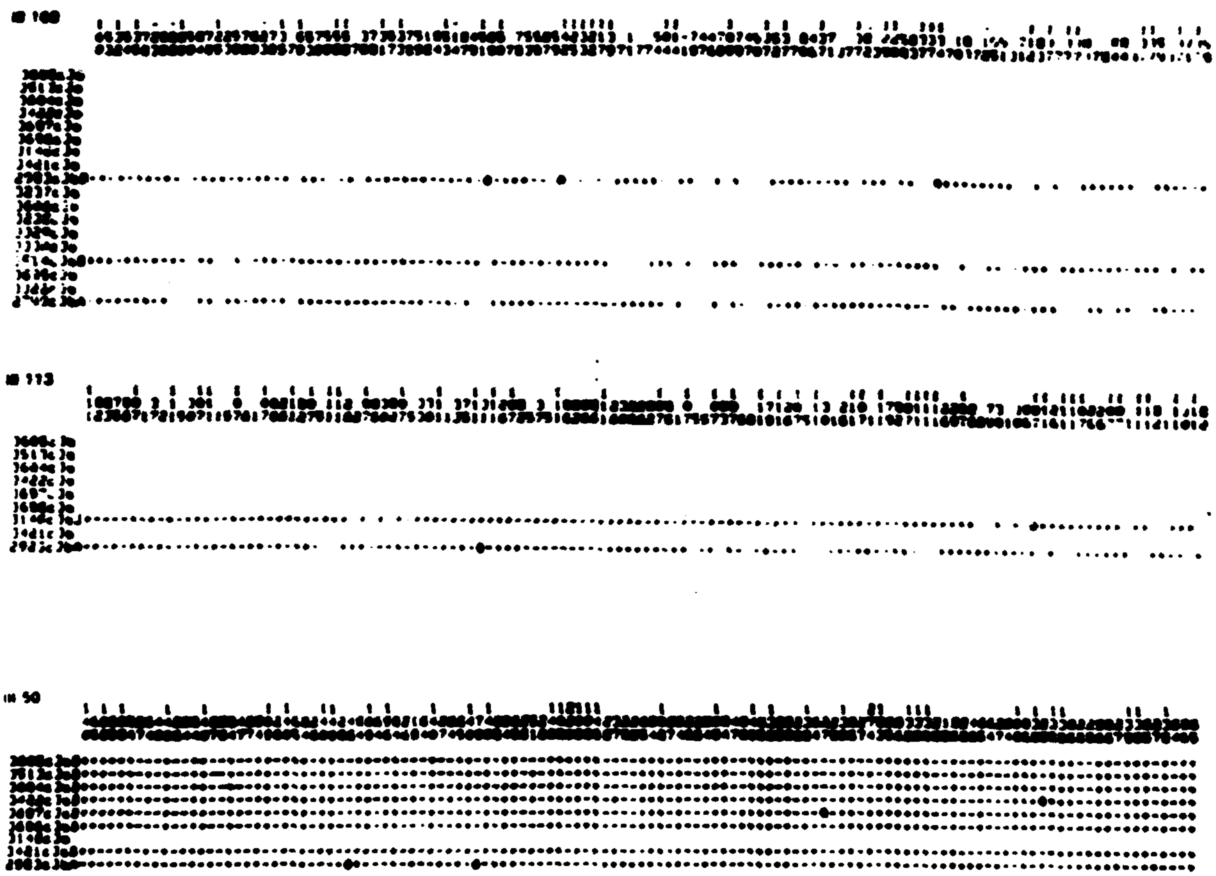


Figure C-10. Case III Update 190

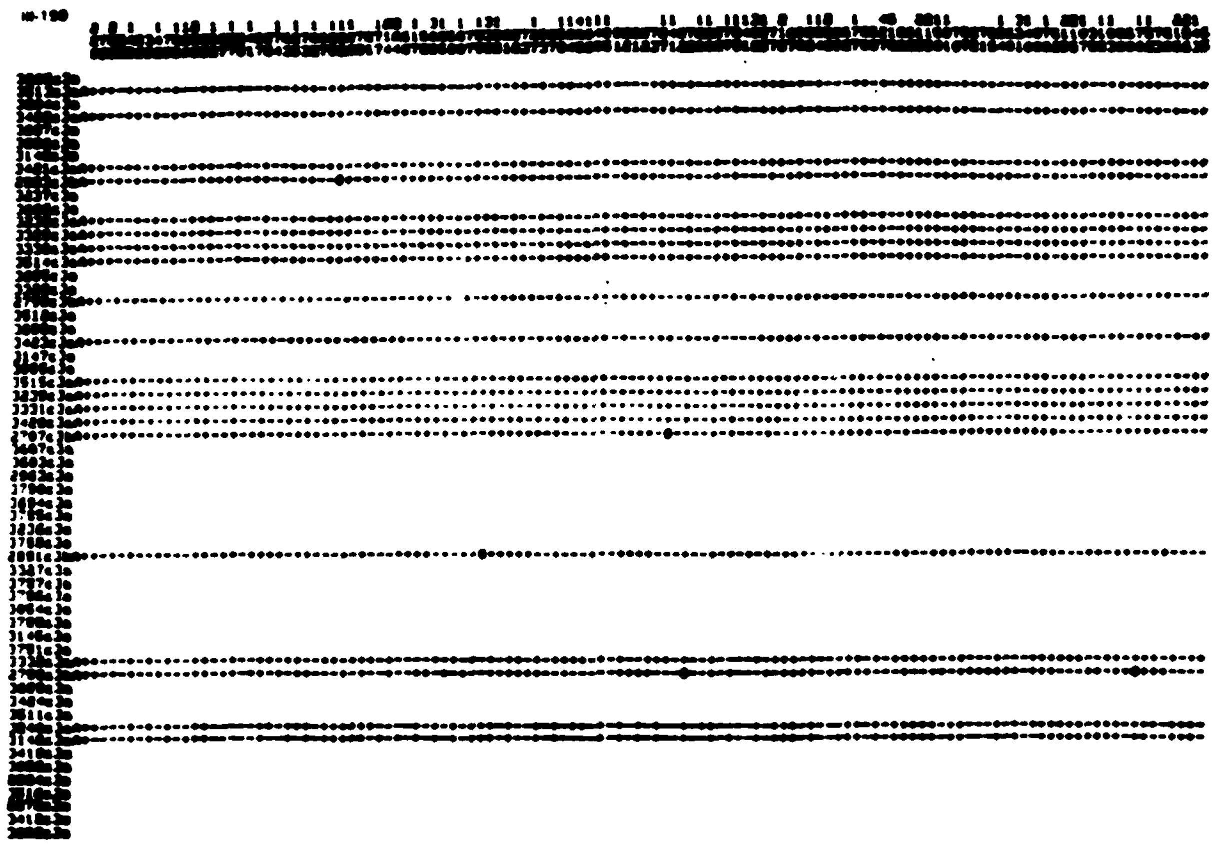


Figure C-11. Case III Update 220

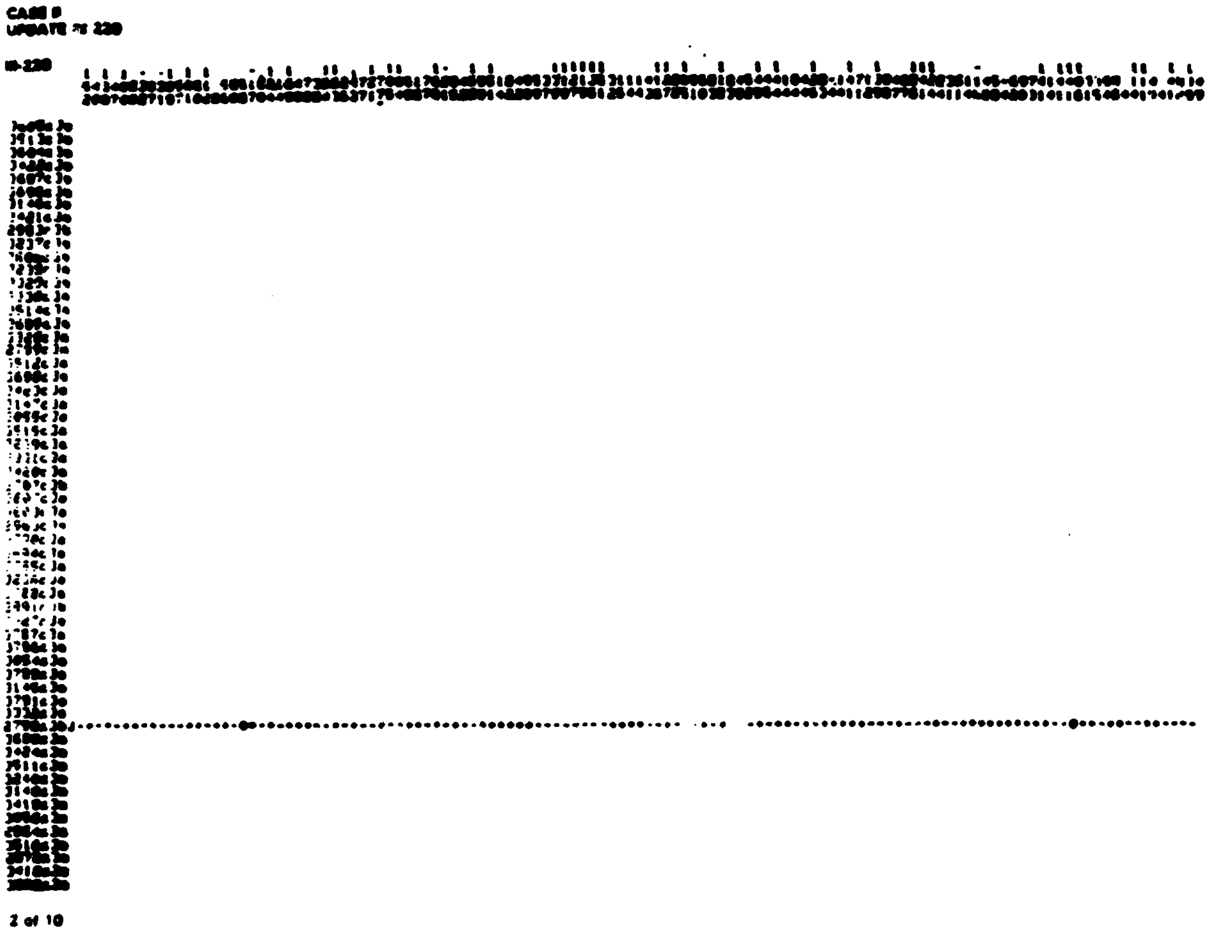


Figure C-12. Case III Update 360

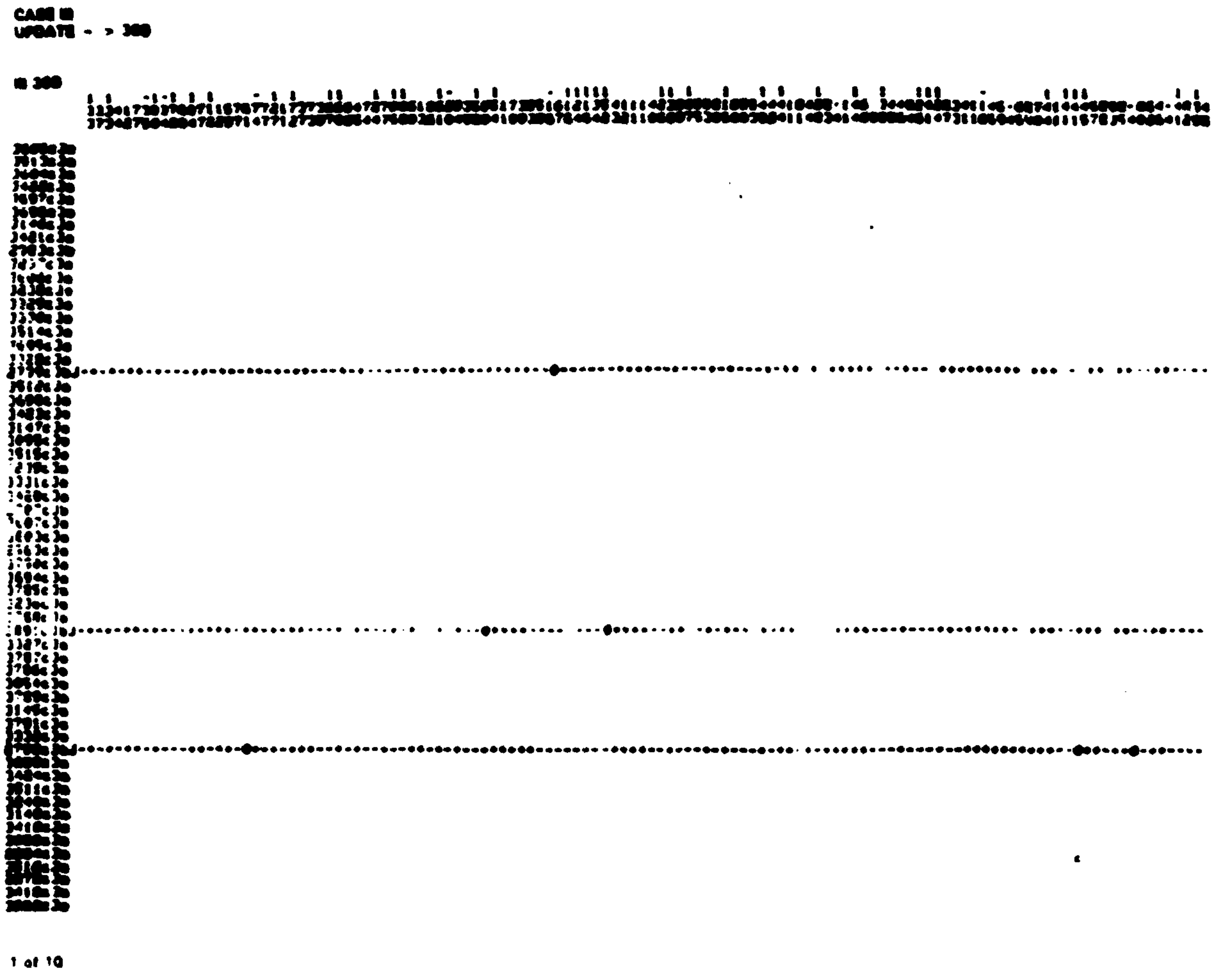
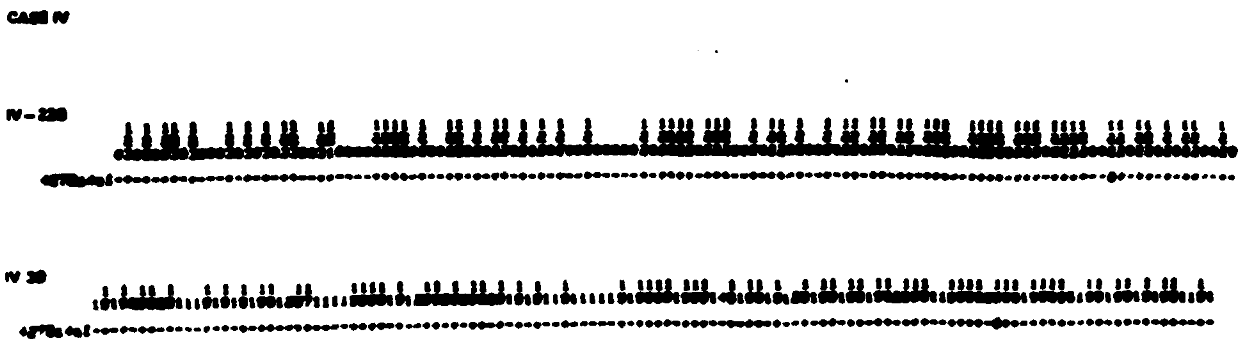


Figure C-13. Case IV Non-convergence



number counts the number of iterations into the algorithm.

C.1 OBSERVATIONS OF CASE I

Case I is represented by Figures C-1, C-5, C-6 and Table C-1. Figure C-5 demonstrates the integrating dead zone sign function (Equation B.18) as the beam weights (Equation. B.8) increase with the sequence: $x(0) = 0$, $x(1) = 1$, $x(2) = 3$, $x(3) = 6$. Figure C-6 details full convergence as all the areas of interest shown meet the constraint 98 iterations into the algorithm. Observation of the beam weights (Table C-1) shows that over half of the beams are dominant. The area definition was chosen from the singlet beam library[4] to center area "C1A" between singlet beams 40, 41, and 48 and area "C2A" directly below singlet beam 16. The algorithm makes no use of major lobe illumination. Table C-1 shows singlet beam 16 as one of the lowest in magnitude.

C.2 OBSERVATIONS OF CASE II

Case II is represented by Figures C-2, C-7, C-8, and Table C-1. This case is representative of typical gain coverage contours. A noteworthy observation (shown in Figures C-7 and C-8) is the failure of the algorithm to satisfy a small subset of area/gain points. Twenty-one points were considered after 180 iterations (approximately 1 hour of execution time). While beam weights were produced, the MBR_AAS did not consider any points from area "C2C".

C.3 OBSERVATIONS OF CASE III

Case III is represented by Figures C-9 through C-12 and Table C-1. This case represents a terminal near the satellite substation point and a jammer inside the ±4° look angle. Distances between terminal and jammer areas are roughly 500 miles. After 2 hours and 20 minutes, the terminal area (C3A) was properly fit but the jammer location (C3B) oscillated in and out of the bounds (note the "J" descriptor on all jammer points).

---

[4]A library of individual singlet beams is maintained by DCEC Code R420

C.4 OBSERVATIONS OF CASE IV

Case IV is represented by Figure C-13 and Table C-1. This case represents a terminal location near the horizon (i.e., a large look angle). This case gives ample insight to the performance of this algorithm. Not one point was satisfied after one hour of execution.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A163 530

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY N/A	3. DISTRIBUTION/AVAILABILITY OF REPORT UNLIMITED
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Final Report 170 Log No. MSO-85-093b	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION M/A-COM LINKABIT, Inc. 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Defense Communications Agency MILSATCOM Systems Office, Code A800
6c. ADDRESS (City, State and ZIP Code) 8619 Westwood Center Drive Vienna, VA 22180	7b. ADDRESS (City, State and ZIP Code) Washington, D.C. 20305
8a. NAME OF FUNDING/SPONSORING ORGANIZATION DCA/DCEC 8b. OFFICE SYMBOL (If applicable) Code R420	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code) 1860 Wiehle Street Reston, Virginia	10. SOURCE OF FUNDING NOS. PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.
11. TITLE (Include Security Classification) DSCS Network Performance Software Support (U)	

12. PERSONAL AUTHOR(S)
S. Van Trees, D. Hughes, S. Huffman, B. Maples, J. Shattuck

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Final	FROM Dec 84 TO 29NOV85	1986 J[illegible]ary 7	130

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB GR	DSCS Network Planning Software (DNPS), Multiple Beam Antenna Algorithms, DSCS Operation Control System (DOCS)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

(U) The purpose of this task was to evaluate the utilization of the DSCS Operational Control System (DOCS) and to provide version release test support for the DSCS Network Planning Software (DNPS). This report also examines the development of multiple beam antenna (MBA) algorithms to calculate beam weights that provide minimum mean square error (MSE) performance between desired and actual gain contour patters.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Mr. Lawrence Krebs	22b. TELEPHONE NUMBER (Include Area Code) (703) 437-2026 22c. OFFICE SYMBOL R420

DD FORM 1473, 83 APR   EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

END

FILMED

3-86

DTIC